

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

THE DESIGN OF A LOCAL AREA NETWORK
CONFIGURATION MANAGEMENT SYSTEM
FOR THE NAVAL POSTGRADUATE SCHOOL
ADMINISTRATIVE SCIENCES DEPARTMENT

by

Douglas A. Suriano

September 1989

Thesis Advisor

Norman F. Schneidewind

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE



REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b RESTRICTIVE MARKINGS	
SECURITY CLASSIFICATION AUTHORITY		3 DISTRIBUTION, AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.	
CLASSIFICATION DOWNGRADING SCHEDULE		5 MONITORING ORGANIZATION REPORT NUMBER(S)	
FORMING ORGANIZATION REPORT NUMBER(S)		7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School	
NAME OF PERFORMING ORGANIZATION Naval Postgraduate School		6b OFFICE SYMBOL (If applicable) 54	
ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000		7b ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	
NAME OF FUNDING / SPONSORING ORGANIZATION		8b OFFICE SYMBOL (If applicable)	
ADDRESS (City, State, and ZIP Code)		9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
		10 SOURCE OF FUNDING NUMBERS	
		PROGRAM ELEMENT NO	PROJECT NO
		TASK NO	WORK UNIT ACCESSION NO
TITLE (Include Security Classification) The Design of a Local Area Network Configuration Management System for the Naval Postgraduate School Administrative Sciences Department			
PERSONAL AUTHOR(S) Mariano, Douglas A.			
TYPE OF REPORT Master's Thesis		13b TIME COVERED FROM TO	14 DATE OF REPORT (Year, Month, Day) 1989 September
		15 PAGE COUNT 116	
SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government			
COSAT CODES		18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP	
ABSTRACT (Continue on reverse if necessary and identify by block number) This thesis identifies system requirements and provides an application system design for an automated Local Area Network (LAN) Configuration Management System to be used by the Naval Postgraduate School Administrative Sciences Department LAN Staff. The proposed system will provide a tool to assist the LAN Staff in carrying out maintenance functions such as tracking changes to the hardware and software configuration of LANs. Problem definition, requirements definition, and system design are among issues addressed. This system is intended for use on a microcomputer based database management system.			
DISTRIBUTION / AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS		21 ABSTRACT SECURITY CLASSIFICATION Unclassified	
NAME OF RESPONSIBLE INDIVIDUAL Professor Norman F. Schneidewind		22b TELEPHONE (Include Area Code) (408) 646-2719	22c OFFICE SYMBOL 54Ss

7245492

Approved for public release; distribution is unlimited.

THE DESIGN OF A LOCAL AREA NETWORK CONFIGURATION
MANAGEMENT SYSTEM FOR THE NAVAL POSTGRADUATE SCHOOL
ADMINISTRATIVE SCIENCES DEPARTMENT

by

Douglas A. Suriano
Captain, United States Marine Corps
B.S., United States Naval Academy

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

ABSTRACT

This thesis identifies system requirements and provides an application system design for an automated Local Area Network (LAN) Configuration Management System to be used by the Naval Postgraduate School Administrative Sciences Department LAN Staff. The proposed system will provide a tool to assist the LAN Staff in carrying out maintenance functions such as tracking changes to the hardware and software configuration of LANs. Problem definition, requirements definition, and system design are among issues addressed. This system is intended for use on a microcomputer based database management system.

TABLE OF CONTENTS

I.	INTRODUCTION	1
	A. ORGANIZATION	1
	B. THESIS BACKGROUND	1
	C. THESIS SCOPE	2
	D. METHODOLOGY	3
II.	PROBLEM DEFINITION	4
	A. BACKGROUND	4
	B. PROBLEM DEFINITION	5
	C. SCOPE	6
	D. PROJECT FEASIBILITY	6
	1. Cost	6
	2. Technical	7
III.	REQUIREMENTS DEFINITION	8
	A. DATA REQUIREMENTS	8
	B. FUNCTIONAL REQUIREMENTS	9
IV.	SYSTEM DESIGN	15
	A. LOGICAL DESIGN	15
	B. APPLICATION DESIGN	17

V. CONCLUSIONS AND RECOMMENDATIONS	20
A. SUMMARY	20
B. SYSTEM IMPLEMENTATION	20
C. SUGGESTIONS FOR FURTHER RESEARCH	21
APPENDIX A SYSTEM DATA DICTIONARY	22
APPENDIX B DATA FLOWS	38
APPENDIX C FUNCTIONAL COMPONENT DESCRIPTIONS	45
APPENDIX D APPLICATION DESIGN COMPONENTS	84
LIST OF REFERENCES	108
BIBLIOGRAPHY	109
INITIAL DISTRIBUTION LIST	110

INTRODUCTION

A. ORGANIZATION

The Administrative Sciences Department (AS) of the Naval Postgraduate School (NPS) is responsible for various Local Area Networks (LAN) which are used for education and research. These computer networks are operated under the supervision of the LAN staff.

B. THESIS BACKGROUND

With the advent of Local Area Network technology and the desire for microcomputer connectivity, LANs have become very popular in recent years. These installed LANs play host to many microcomputers and assorted peripherals. A problem lies in managing the configuration of this equipment and required software. Currently, in the AS Department at NPS, LAN configuration management is being accomplished manually. This method tends to be cumbersome and inefficient. An automated system, using existing microcomputer database technology, would simplify recording, updating and deleting data that must be maintained.

As stated in NAVPGSCOLINST 5239.1, the departmental ADP System Security Officers (ADPSSO) are responsible for maintaining an inventory of all ADP hardware, peripherals, and

commercial software in the assigned area [Ref. 1]. An automated LAN Configuration Management System would assist the ASPSSO in fulfilling his responsibilities.

Along with maintaining an accurate hardware and software inventory, the LAN administrator is responsible for overall maintenance of the LANs. Such maintenance may include modification to the hardware and software configurations of a LAN due to equipment failure, hardware upgrades, or software improvements. In order to perform this function adequately, the administrator must know the current configuration before changes can be made to correct a problem. After changes are made they must be documented. As with software maintenance, the goal of this process is to achieve a high degree of traceability, or the ability to trace through all relevant information that may assist the maintainer [Ref. 2]. Again, a LAN Configuration Management System could provide a tool to aid the LAN staff with its maintenance responsibilities.

C. THESIS SCOPE

This thesis will identify system requirements and provide an application system design for an automated Local Area Network Configuration Management System. This will include: determining functional requirements, determining data requirements, developing database design, and developing the application design.

D. METHODOLOGY

The traditional approach to systems development will be utilized to complete this thesis. It consists of a definition phase, a requirements phase, and a design phase. The AS Department LAN staff at NPS will act as the users, that is, the group that will benefit from this design. They will help to identify functions and data elements that are required for this system.

The design methodology used to complete this thesis was taken from Kroenke and Dolan's [Ref. 3] "Database Processing: Fundamentals, Design, Implementation." This methodology focuses on the object-oriented approach to database design. An object is defined by Kroenke and Dolan as "a stored data representation of an entity of concern to the user." An example of an object is a local area network. Although a LAN is made up of many different parts, it can be viewed as a single object. These objects can be used to derive relations, forms, reports, and even application design.

II. PROBLEM DEFINITION

A. BACKGROUND

The AS Department LAN staff is responsible for the administration and management of the LANs listed below.

1. IBM PC Network (RM 250)
2. IBM Token Ring Network (RM 224)
3. 3COM Ethernet (RM 224)
4. Appletalk Network (RM 224)
5. IBM Token Ring Network (RM 158)
6. TOPS Network (RM 224)

The management of these LANs encompasses a wide range of functions. The Open Systems Interconnect (OSI) model defines five areas of network management: fault management, performance analysis, security, accounting and configuration management [Ref 4]. It is the responsibility of the LAN staff to address these areas of LAN management.

One of the more difficult functions of managing a LAN is configuration management. Configuration management is the system of tracking changes to hardware and software associated with the network. This tracking function is linked closely to the traceability issue discussed earlier.

Some of the hardware that can be tracked may include: server and user computers, network boards, modems, memory

boards, peripherals, and cable plant components. Not only is it important to track the hardware inventory, but also the various settings that may be associated with each individual piece of hardware. For example, the switch settings on a network board would be tracked and documented to reflect any problems that may have occurred with interrupt level conflicts. The task of tracking software would include application software on server and user machines, as well as network programs.

Currently, the configuration management of the AS LANs is being accomplished with the aid of a word processor and three-ring binders. The hardware and software inventories are recorded by using the word processor program. The information is then entered, saved to disk and printed to hard copy for storage in the binders. The binders are kept in the room in which the LAN is located. When a change in the configuration of hardware or software is made, a pen and ink change is made to the hard copy located in the binder and periodically updated using the word processor program.

B. PROBLEM DEFINITION

It is the responsibility of the AS LAN Staff to perform the following configuration management functions.

1. Maintain accounting of all AS LAN hardware and software.
2. Maintain the hardware configuration of individual LAN computers.

3. Maintain the software configuration of individual LAN computers.

The current configuration management system being used by the AS lab staff is difficult to keep updated. Because the system is not very user friendly, proper updates to the documentation are often delayed or even neglected.

C. SCOPE

The AS LAN Staff is requesting an automated information system that will:

1. Reside on an existing server.
2. Provide a mechanism for tracking the hardware and software configuration of all existing AS Department LANs.

D. PROJECT FEASIBILITY

When assessing the feasibility of a proposed system, two areas must be addressed; the cost of implementing and operating the new system and the technical limitations to be considered.

1. Cost

The application software and hardware necessary to implement the design of this system are available in the AS Department. The design and follow-on programming will be accomplished by students or staff at NPS as a part of thesis requirements and normal LAN maintenance.

2. Technical

The requested system is well suited for a personal computer based, relational database application. With the existing hardware and database software technology available to the AS Department, a database application such as the one proposed in this thesis is well within technical constraints.

III. REQUIREMENTS DEFINITION

A. DATA REQUIREMENTS

The following objects are identified as necessary in defining the data that is to be captured in the configuration management system (see Appendix A). Figures A.4a and A.4b show the data elements for the LANs.

1. LAN object contains data that uniquely identifies a specific LAN.
2. NODE object contains essential data on individual computers attached to a specific network.
3. ACCESSORY object contains essential data on accessories, such as modems and emulation boards, that may be attached to an individual computer. There may be several different accessories attached to one individual computer.
4. SETTING object contains essential information on the way the switch settings are configured for a specific accessory. Comments documenting these settings are also addressed in this object.
5. APPLICATION-SOFTWARE object contains essential information on application software that is loaded on an individual node.
6. SYSTEM-SOFTWARE object contains essential data on system software, such as DOS or the PC LAN operating system, that are loaded on an individual node.
7. CABLE-PLANT object contains essential data on the actual cable plant of the network. Items captured in this object may include multiple access units (MAU) and twisted pair cable.
8. SPARE-PARTS object contains data on the spare parts inventory being maintained by the LAN staff.

The objects listed above were derived by examining inventory sheets that are currently being used by the AS LAN staff. After examining the current system and receiving input from the LAN staff, the object definitions, domain definitions, and properties were established (see Appendix A).

B. FUNCTIONAL REQUIREMENTS

The AS LAB Staff is responsible for all data flows into and out of the system. Based on changes in the configuration of a LAN, the staff will add, update, and delete information contained in the database, query the database, and generate necessary reports. Appendix B contains the logical data flow diagrams (DFD) which illustrate these information flows.

The processes necessary to perform the required add functions to the data are shown in Table 1.

TABLE 1

ADD FUNCTIONS

<u>Function</u>	<u>Description</u>	<u>Reference</u>
Add NODE data	Used to add new instance of server or user computer to the data store.	Figure C.1
Add ACCESSORY data	Used to add a new instance of an accessory piece of hardware such as a modem, network board or printer that is attached to a node to the data store.	Figure C.2
Add SETTING data	Used to add a new instance of a switch block setting that is associated with a particular accessory in a node to the data store.	Figure C.3
Add APPLICATION-SOFTWARE data	Used to add a new instance of application software that is loaded into a node to the data store.	Figure C.4
Add SYSTEM-SOFTWARE data	Used to add a new instance of system software that is running on the LAN to the data store.	Figure C.5
Add CABLE-PLANT data	Used to add a new instance of the CABLE-PLANT object to the data store.	Figure C.6
Add SPARE-PARTS data	Used to add a new instance of the SPARE-PARTS object to the data store.	Figure C.7

The processes necessary to perform the required update functions to the data are shown in Table 2.

TABLE 2

UPDATE FUNCTIONS

Function	Description	Reference
Update NODE data	Used anytime a change is required to any data in the NODE data store.	Figure C.8
Update ACCESSORY data	Used anytime a change is required to any data in the ACCESSORY data store.	Figure C.9
Update SETTING data	Used anytime a change is required to any data in the SETTING data store.	Figure C.10
Update APPLICATION-SOFTWARE data	Used anytime a change is required to any data in the APPLICATION-SOFTWARE data store.	Figure C.11
Update SYSTEM-SOFTWARE data	Used anytime a change is required to any data in the SYSTEM-SOFTWARE data store.	Figure C.12
Update CABLE-PLANT data	Used anytime a change is required to any data in the CABLE-PLANT data store.	Figure C.13
Update SPARE-PARTS data	Used anytime a change is required to any data in the SPARE-PARTS data store.	Figure C.14

The processes necessary to perform the required delete functions to the data are shown in Table 3.

TABLE 3

DELETE FUNCTIONS

Function	Description	Reference
Delete NODE data	Used when a node is removed from a network to remove that particular instance from the NODE data store. The associated ACCESSORY and SETTING instance for that node will also be deleted.	Figure C.15
Delete ACCESSORY data	Used when an accessory is removed from a node to remove that particular instance from the ACCESSORY data store. The associated SETTING instance for that accessory will also be deleted.	Figure C.16
Delete APPLICATION-SOFTWARE data	Used anytime application software is removed from a node to remove an instance from the APPLICATION-SOFTWARE data store.	Figure C.17
Delete SYSTEM-SOFTWARE data	Used anytime system software is removed from the network to remove an instance from the SYSTEM-SOFTWARE data store.	Figure C.18
Delete CABLE-PLANT data	Used anytime an item of cable plant is removed from a network to remove an instance from the CABLE-PLANT data store.	Figure C.19
Delete SPARE-PARTS data	Used anytime a spare part is removed from the spare parts inventory to remove an instance from the SPARE-PARTS data store.	Figure C.20

Outputs from the system are produced by utilizing the query and print functions of the system. The query function will enable the user to display desired information to the screen. The print function will enable the user to print desired hard copy reports on a printer.

The processes necessary to perform required query functions on the data are shown in Table 4.

TABLE 4
QUERY FUNCTIONS

<u>Function</u>	<u>Description</u>	<u>Reference</u>
Query NODE data	Used anytime the LAN staff wants to view data on a particular node on the network.	Figure C.21
Query ACCESSORY and SETTING data	Used anytime the LAN staff wants to view data on a particular accessory and associated switch settings.	Figure C.22
Query APPLICATION/ SYSTEM-SOFTWARE data	Used anytime the LAN staff wants to view software associated with a particular node.	Figure C.23
Query SYSTEM- SOFTWARE data	Used anytime the LAN staff wants to view only system software that is associated with a particular network.	Figure C.24
Query CABLE-PLANT data	Used anytime the LAN staff wants to view the cable plant items associated with a particular network.	Figure C.25
Query SPARE-PARTS data	Used anytime the LAN staff wants to view the spare parts items in the spare parts inventory.	Figure C.26

The output reports function of the system are shown in Table 5.

TABLE 5
REPORT FUNCTIONS

<u>Function</u>	<u>Description</u>	<u>Reference</u>
Node Hardware Configuration	Derived from data contained in the NODE and ACCESSORY data stores.	Figure C.27
Node Software Configuration	Derived from data contained in the SYSTEM APPLICATION-SOFTWARE data stores.	Figure C.28
System Software Configuration	Derived from data contained in the SYSTEM-SOFTWARE data store.	Figure C.29
Cable Plant Configuration	Derived from data contained in the CABLE-PLANT data store.	Figure C.30
Spare Parts Listing	Derived from data contained in the SPARE-PARTS data store.	Figure C.31

Appendix C contains a summary of the Add, Update, Delete, and Display mechanisms required for each of the data stores and the special control mechanisms required to fulfill requirements of the LAN staff. There are no additional inter-application, operational, administrative, or environmental requirements to be addressed.

IV. SYSTEM DESIGN

A. LOGICAL DESIGN

This application is comprised of eight relations as shown in Figure D.1. The LAN relation contains the identification number data of each specific LAN. The NODE relation contains all of the individual node data that is common across all of the networks. The ACCESSORY relation contains data on all of the hardware accessories that may be attached to or plugged into a specific node. The SETTING relation contains switch setting data and comments about a specific accessory that is associated with an individual node. The APPLICATION-SOFTWARE relation contains data on the specific application software that are loaded on a particular node. The SYSTEM-SOFTWARE relation contains data on each operating system that is running on a particular network. The CABLE-PLANT relation contains data on the different cable plant hardware that connects the network. The SPARE-PARTS relation contains data on the spare parts that are currently held in inventory. The relation definitions of each of the above mentioned relations are shown in Appendix D.

Seven basic relationships exist in this application to satisfy the desired functional requirements of the user. A one-to-many optional relationship exists between the LAN

object and two of its major components: the NODE object, and the CABLE-PLANT object. A one-to-many mandatory relationship exists between the LAN object and the its remaining major component, the SYSTEM-SOFTWARE object. These relationships exist because each component has a specific LAN with which it is associated. A LAN is made up of three major components: nodes, which are user and server computers, system software, such PC LAN and DOS operating systems, and cable plant hardware, such as cable and connectors which connect the components to the network. A one-to-many, optional relationship between the NODE object and two of its subcomponents, the ACCESSORY object and APPLICATION-SOFTWARE object, exists because each node has individual accessories, such as network boards, and modems, that are attached to each node, and individual application software that are installed on each node. A many-to-many relationship exists between the NODE object and the SYSTEM-SOFTWARE object because each node must have system software installed to be attached to the network. A node may have several types of system software installed. A final relationship exists between the ACCESSORY object and the SETTING object because each accessory may have switch settings and comments that are required to be documented to allow the LAN staff to manage special situations such as interrupt level conflicts that may arise when a new accessory is added to a node. No relationship exists between

the SPARE-PARTS object and any other object because the spare parts inventory is independent of any specific LAN.

Some constraints must be recognized in this application. First, an optional relationship exists between the LAN object and two of its components: the NODE object and the CABLE-PLANT object. That is, each LAN is not required to have specific nodes or cable plant hardware, but each component must be associated with a LAN. Likewise, an optional relationship exists between the NODE object and two of its subcomponents: the ACCESSORY object and the APPLICATION-SOFTWARE object. That is, each node is not required to have specific accessories attached to, or software applications loaded on it. However, each subcomponent must be associated with a node. The same can be said for the relationship that exists between the ACCESSORY object and the SETTING object. The SYSTEM-SOFTWARE object has a mandatory relationship with the LAN object and NODE object because the LAN must have system software to exist and the various nodes must have system software installed to be attached to the network. A graphical representation of these relationships is depicted in Figure D.1.

B. APPLICATION DESIGN

This application is designed as a Multi-Application, Single-User Relational Database System to be used on a stand-

alone microcomputer. The menu driven interface design is intended to minimize training time and make the system easy to use. The menu hierarchy is shown in Figure D.2.

An Object/Function orientation was used to develop the menu structure. Different objects were identified to ensure that all required data elements would be captured and various functions that would use these objects were identified to satisfy the information needs of the user.

The main menu is shown in Figure D.3. From this menu the user will select a LAN that required additions, changes or deletions or the spare parts option if changes occurred in the spare parts inventory. Once the user has selected a LAN, a LAN menu like the one in Figure D.4 would appear where the user will be prompted to chose a function: add, update, delete, query, or print. Figures D.5 and D.6 show the add options, Figures D.7 and D.8 show the update options, Figures D.9 and D.10 show the delete options, Figures D.11 and D.12 show the query options, and Figures D.13 and D.14 show the print options. The spare parts menu in Figure D.15 would appear if the spare parts option was selected from the main menu.

Appendix D contains the logic necessary to implement each of the low level menu choices, the materializations necessary within each choice, and any integrity/security considerations that must be followed to insure accuracy within the database.

Each of the low level menu choices represent a separate module within the system and should be implemented as such. A menu module should be the driver with which the user will interact to access the various functional modules.

V. CONCLUSIONS AND RECOMMENDATIONS

A. SUMMARY

The LAN Configuration Management System is designed as an interactive, menu-driven system consisting of seven separate, functionally independent applications which access one relationally structured database. Each application will enable the LAN staff to track the configuration of the individual LANs and maintain an accurate spare parts inventory. The system is designed in such way that new LANs may be added with minimal effort as the AS Department acquires additional networks.

B. SYSTEM IMPLEMENTATION

The LAN Configuration Management System is designed to be implemented on a microcomputer based relational database management system, Ashton-Tate's dBASE III PLUS for example [Ref 5]. Implementation should begin by examining the overall structure of the system using the structure charts, object diagrams, object definitions, and domain definitions provided in Appendix A. Next, an understanding of the required functions and data flows shown in Appendix B, will allow for identification of the processes and data that make up the system. Finally, examination of the mechanisms and the

materializations they produce, as depicted in Appendix C along with the menu hierarchy, relation diagram, relation definitions, and application logic shown in Appendix D, will provide a foundation for producing the source code to enable the system to operate.

C. SUGGESTIONS FOR FURTHER RESEARCH

This thesis has produced a design for the NPS Administrative Sciences Department LAN Configuration Management System. To make the system functional, this system design will need to be carried one step further through the implementation phase. The implementation will require a thorough knowledge of database design and a working knowledge of the database management package selected. Further discussion with the user is not necessary, however, it is encouraged. Good communication with users during all phases of systems development is desirable.

APPENDIX A

SYSTEM DATA DICTIONARY

Within the following pages the reader is provided with various items to assist in understanding the physical structure of the LAN Configuration Management System. The contents of this appendix are:

1. System Structure Charts
2. System Object Diagrams
3. System Object Definitions
4. System Domain Definitions

Figure A.1 through Figure A.3 depict the physical structure of the system. The remaining system structure charts such as IBM Token Ring and TOPS Network use the same structure for their respective networks. Those structure charts were omitted to avoid redundancy.

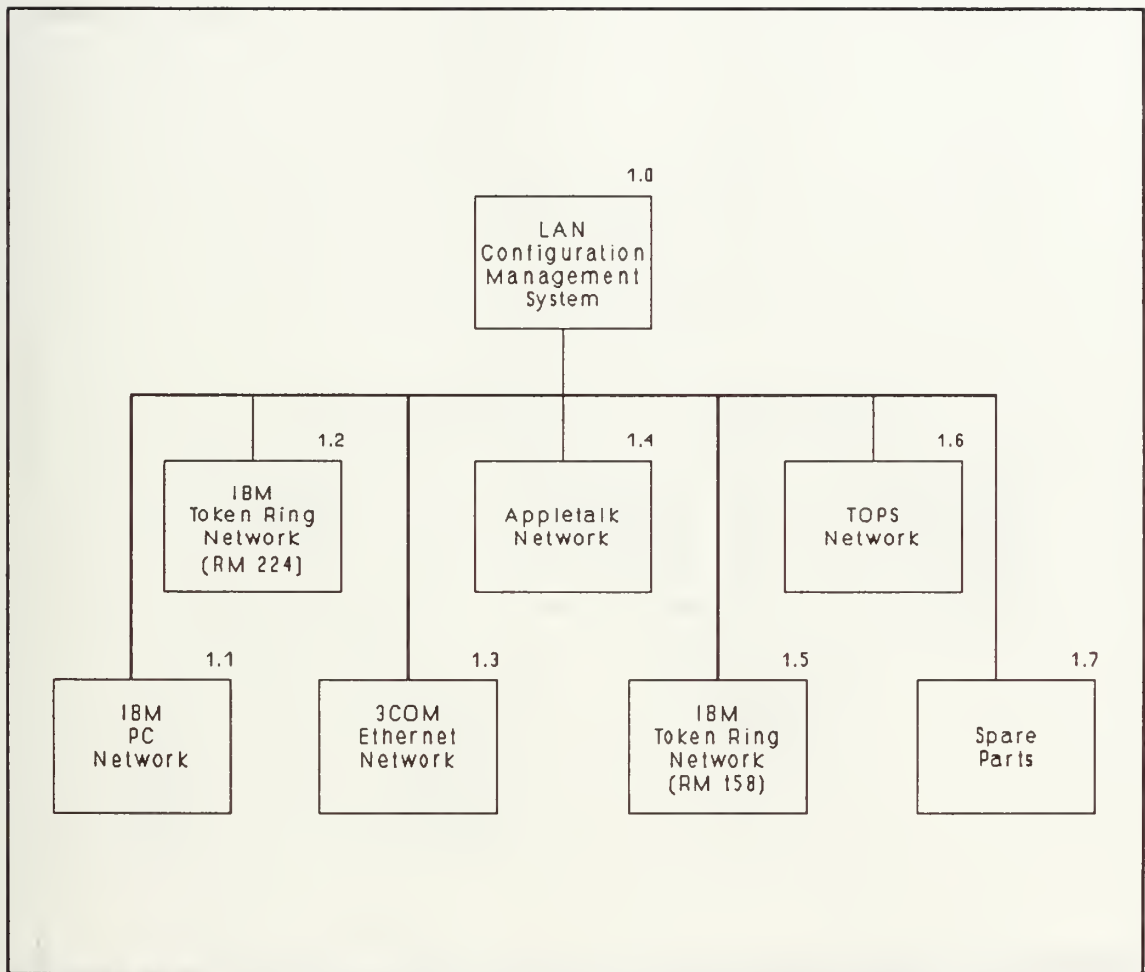


Figure A.1 Level 0 Structure Chart

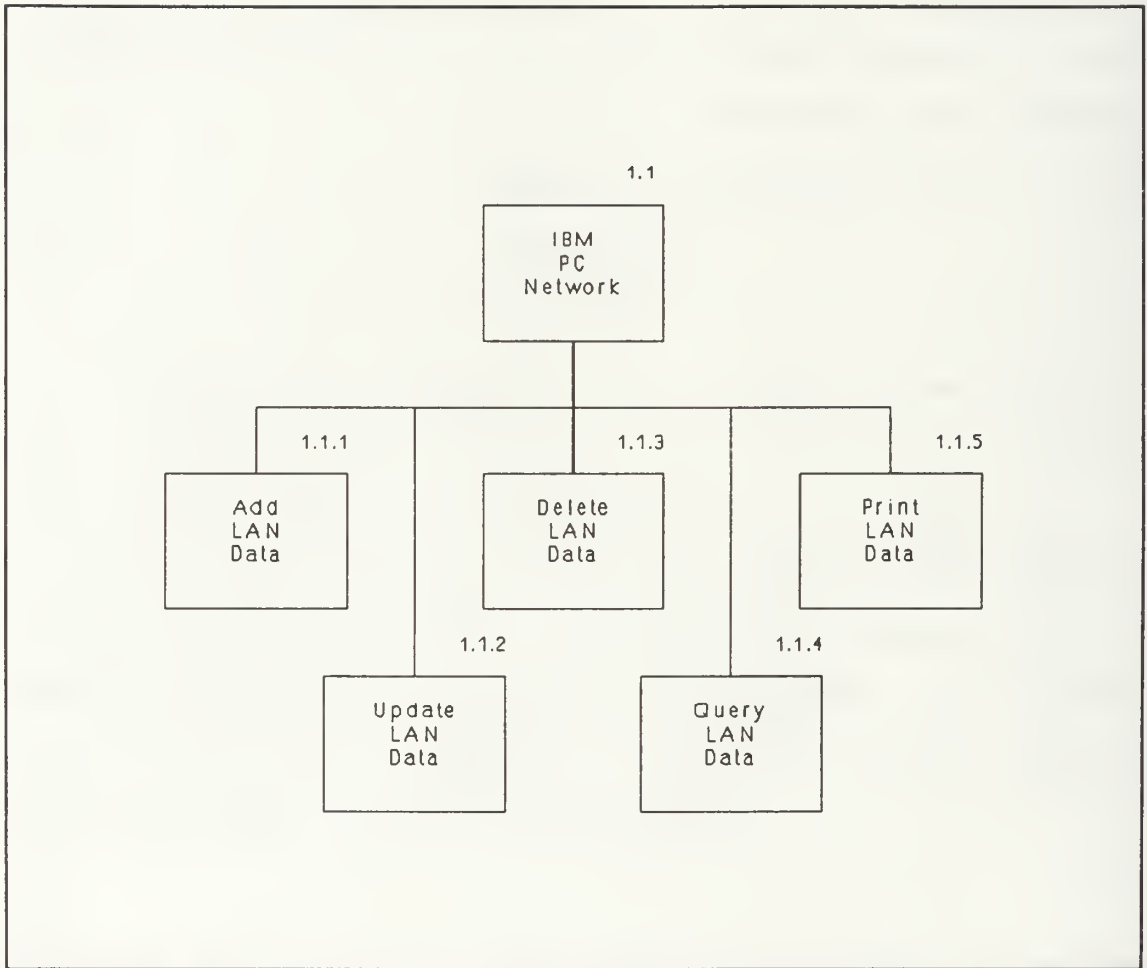


Figure A.2a IBM PC Network Structure Chart

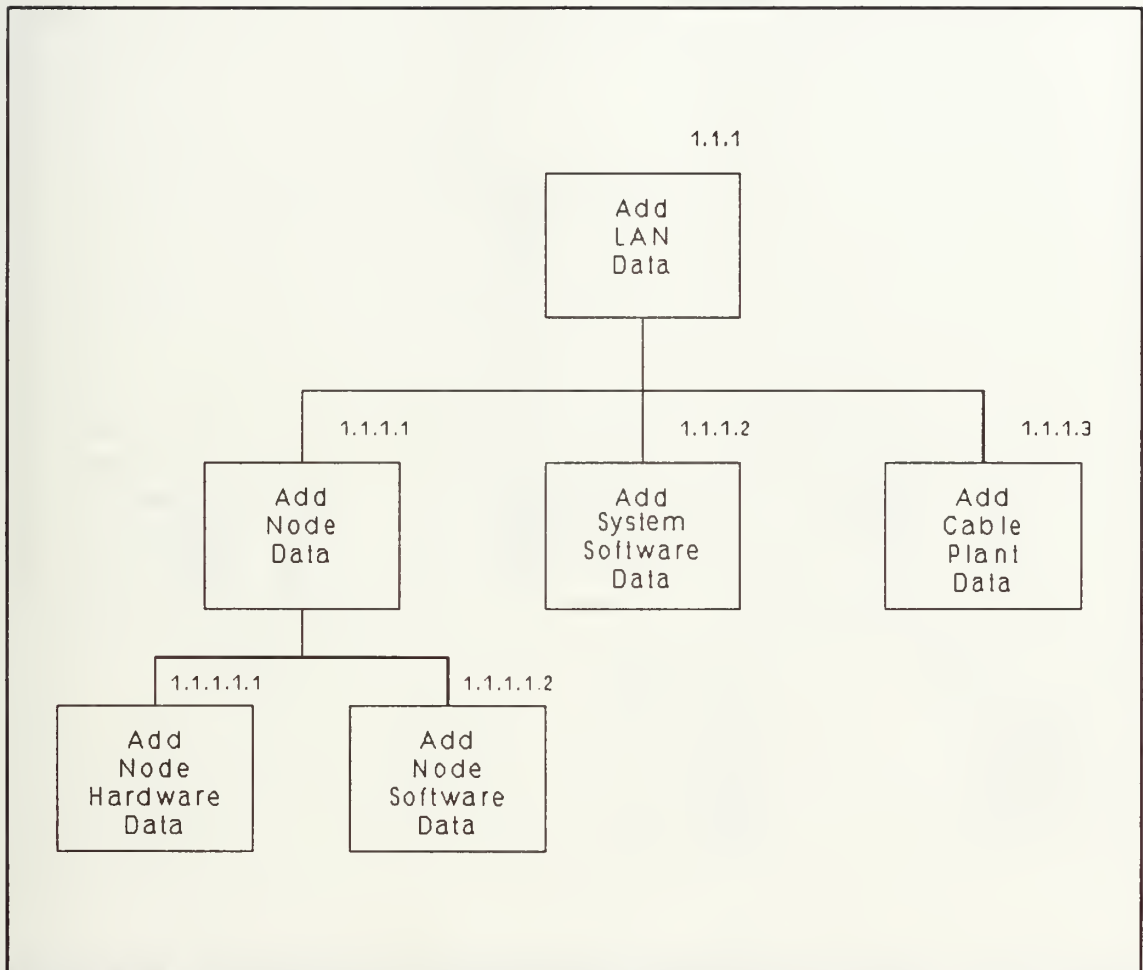


Figure A.2b IBM PC Network Structure Chart (con't)

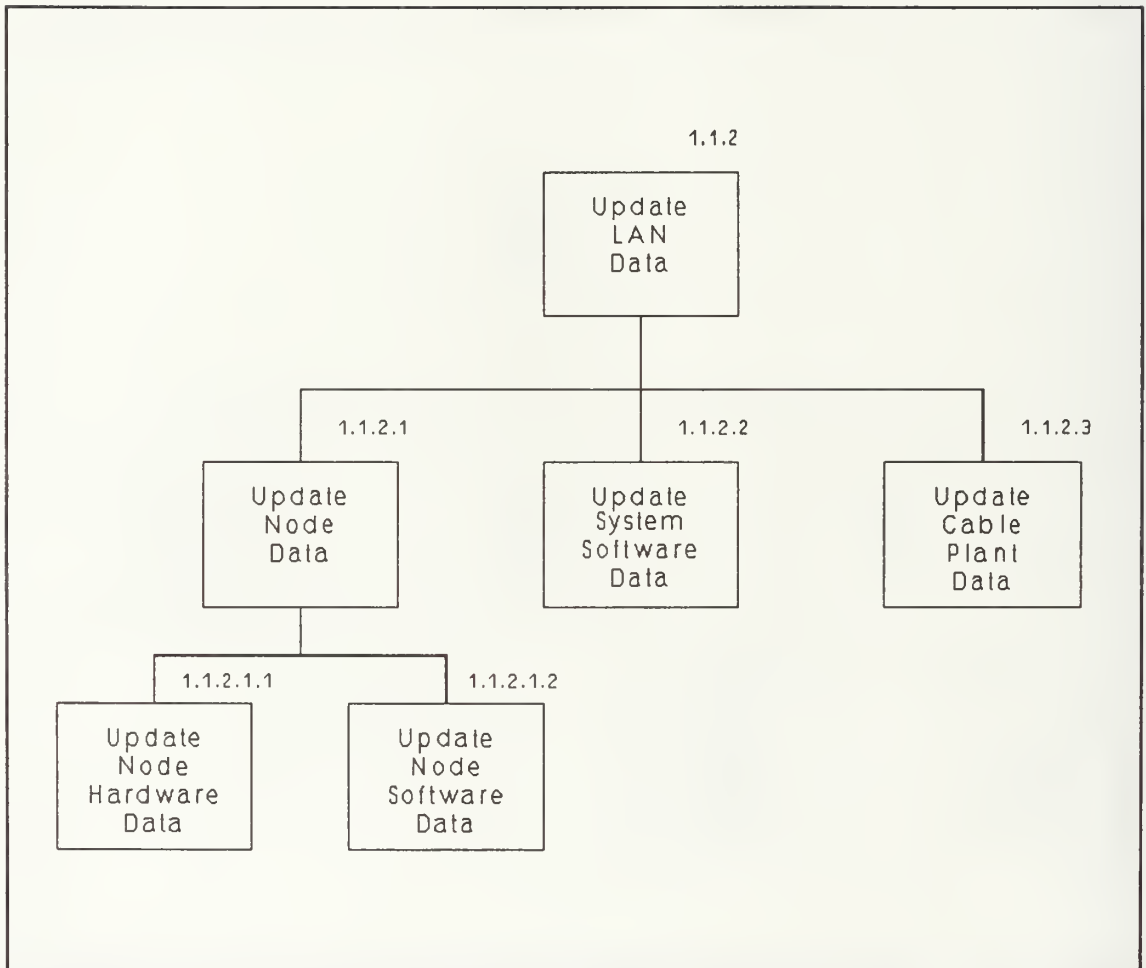


Figure A.2c IBM PC Network Structure Chart (con't)

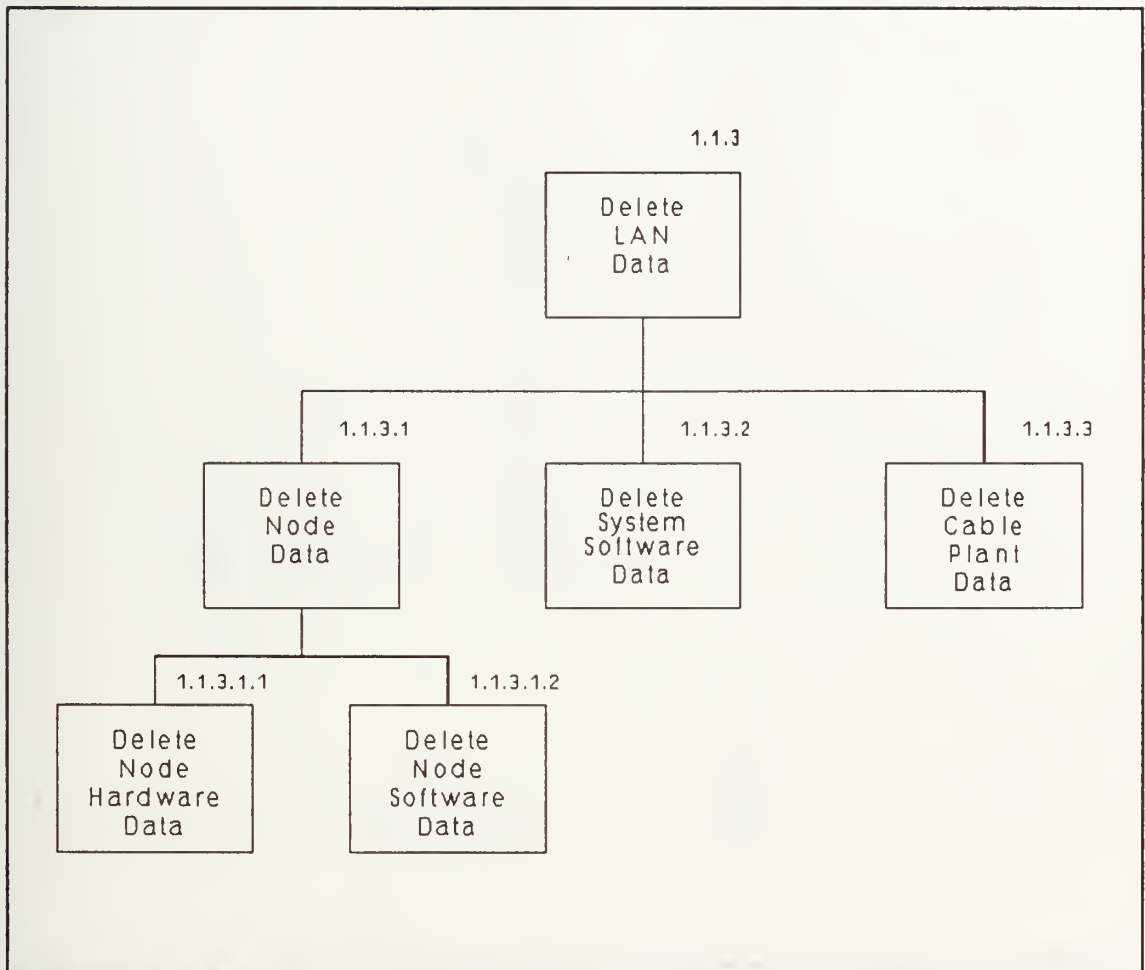


Figure A.2d IBM PC Network Structure Chart

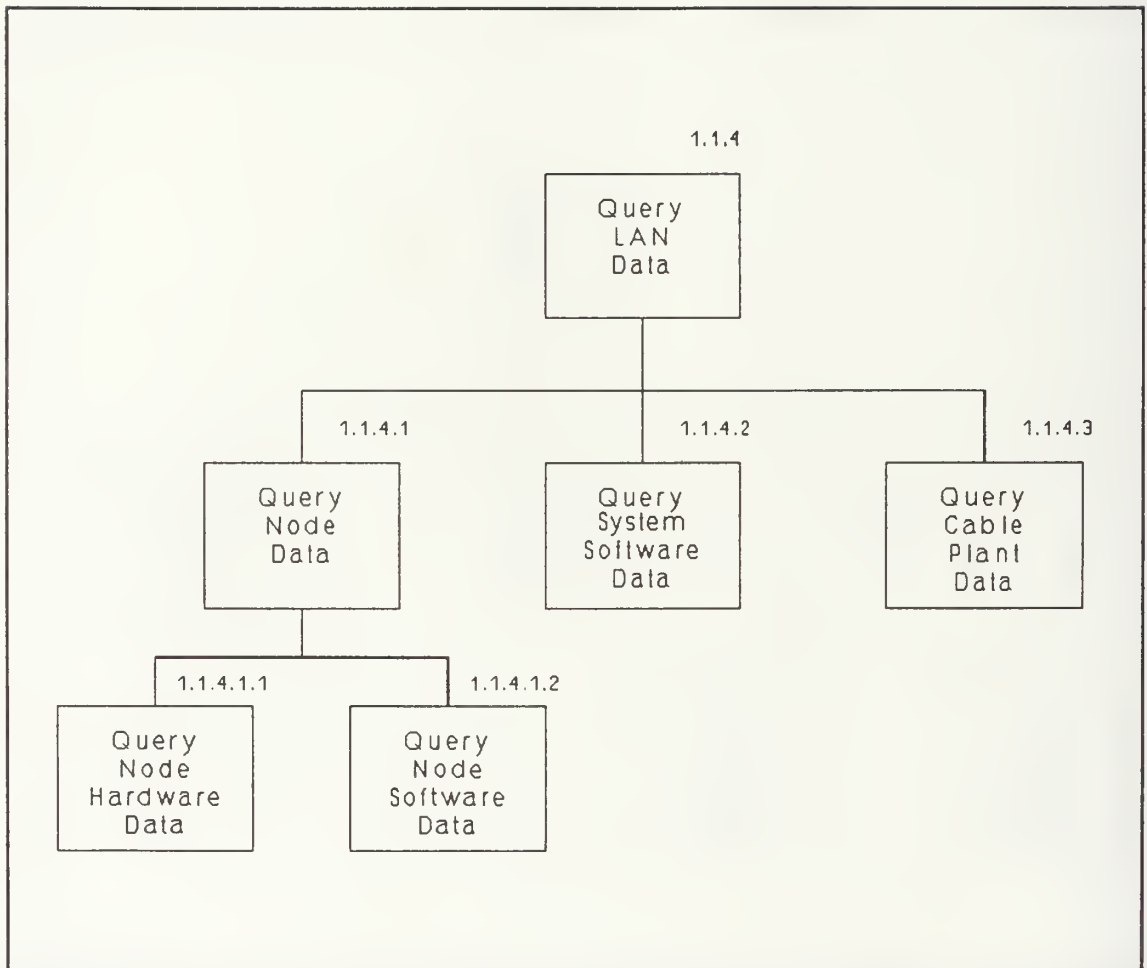


Figure A.2e IBM PC Network Structure Chart (con't)

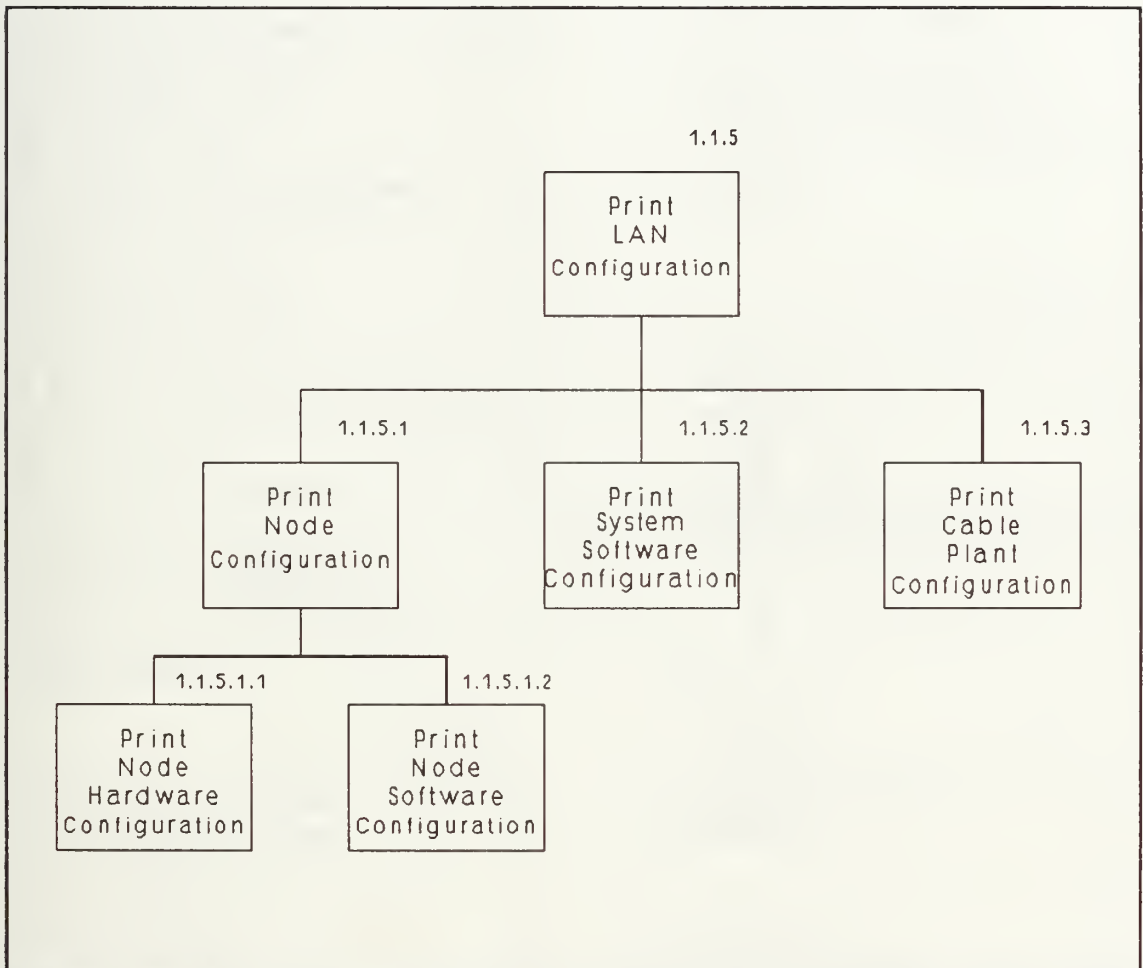


Figure A.2f IBM PC Network Structure Chart (con't)

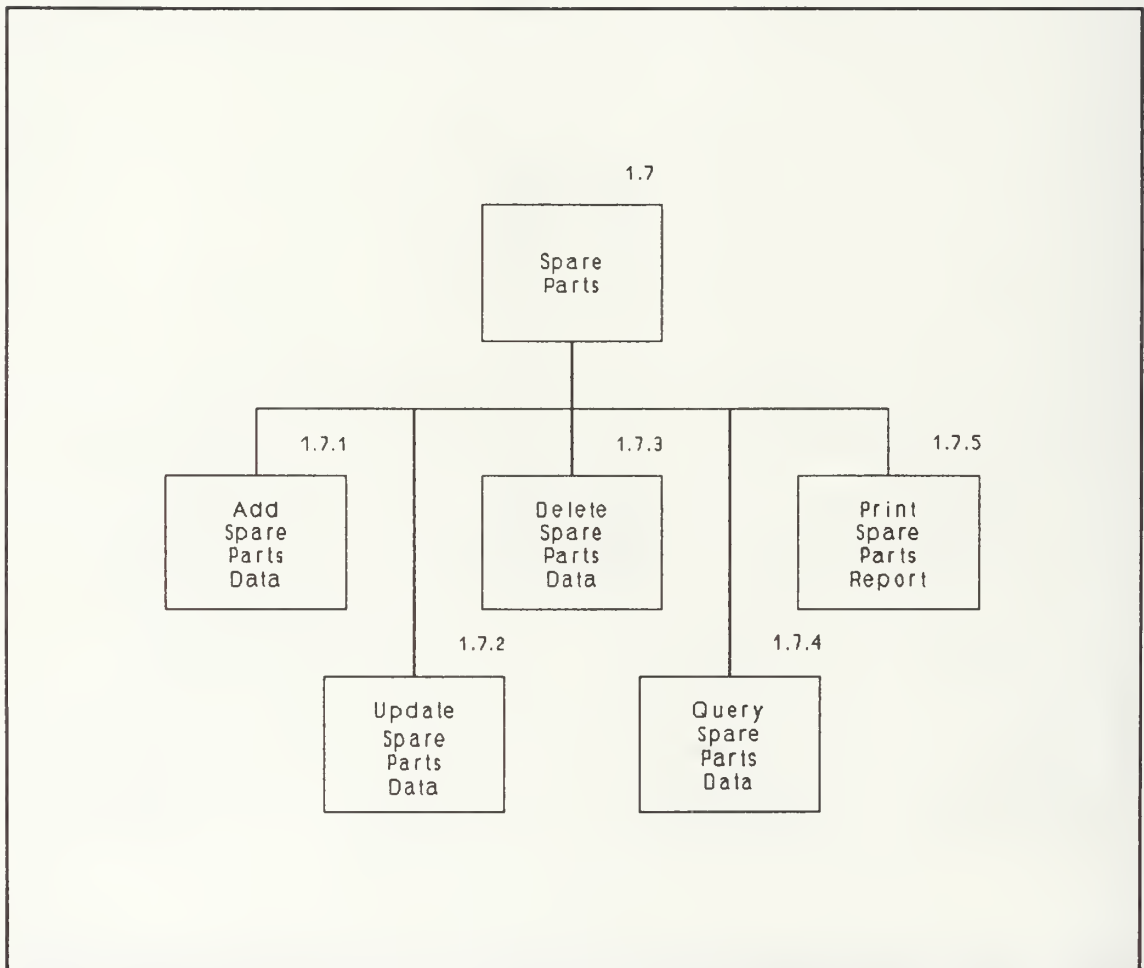
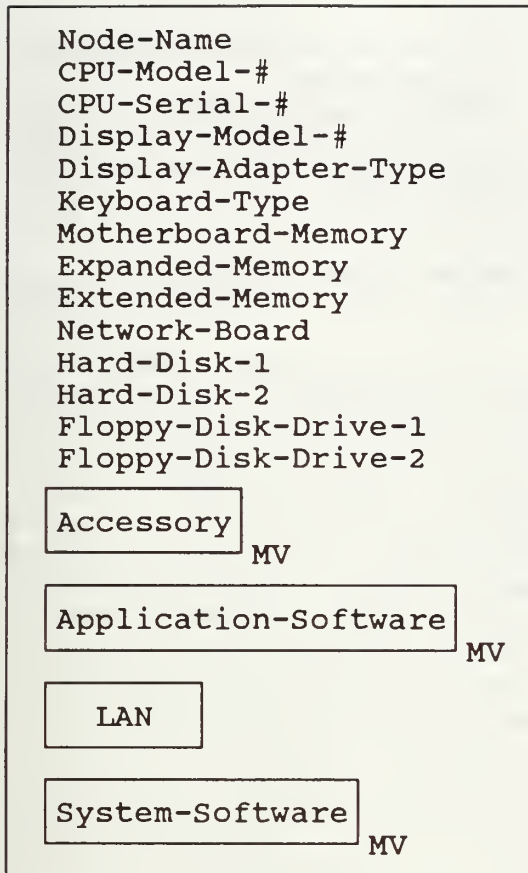
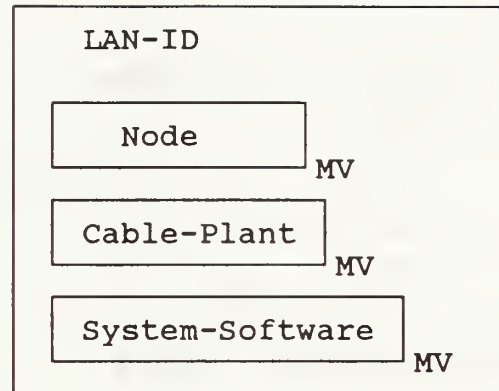


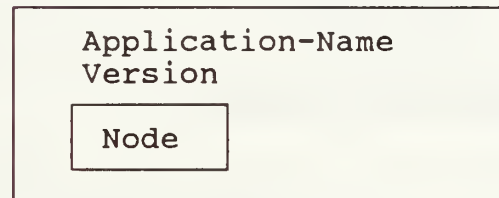
Figure A.3 Spare Parts Structure Chart



NODE OBJECT



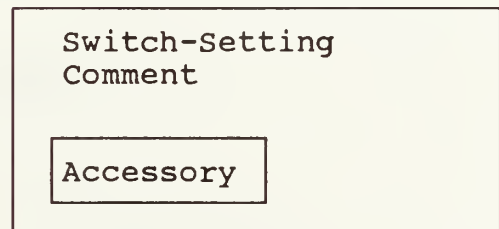
LAN OBJECT



APPLICATION-SOFTWARE OBJECT

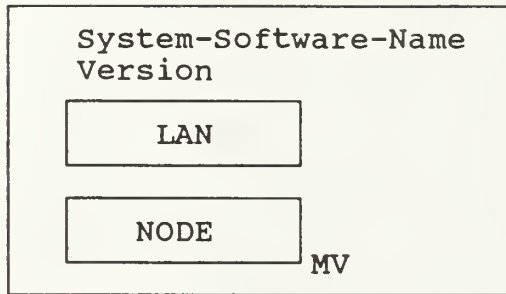


ACCESSORY OBJECT

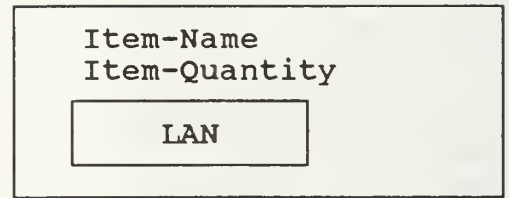


SETTING OBJECT

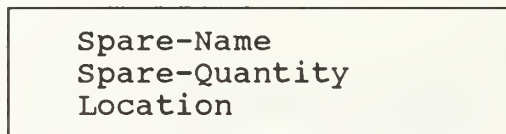
Figure A.4a System Object Diagrams



SYSTEM-SOFTWARE OBJECT



CABLE-PLANT OBJECT



SPARE-PARTS OBJECT

Figure A.4b System Object Diagrams (con't)
OBJECT DEFINITIONS

LAN OBJECT

LAN-ID; LAN-identification
NODE; NODE object; MV
CABLE-PLANT; CABLE-PLANT object; MV
SYSTEM-SOFTWARE; SYSTEM-SOFTWARE object; MV

NODE OBJECT

Node-Name; Computer-node-names
CPU-Model-#; CPU-model-numbers
CPU-Serial-#; CPU-serial numbers
Display-Model-#; Display-model-numbers
Display-Adapter-Type; Display-adapter-types
Keyboard-Type; Keyboard-types
Motherboard-Memory; Motherboard-memory
Expanded-Memory; Expanded-memory
Extended-Memory; Extended-memory
Network-board; Network-board
Hard-Disk-1; Hard-disk-number-one
Hard-Disk-2; Hard-disk-number-two
Floppy-Disk-Drive-1; Floppy-disk-drive-number-one
Floppy-Disk-Drive-2; Floppy-disk-drive-number-two
ACCESSORY; ACCESSORY object; MV
LAN; LAN object
APPLICATION-SOFTWARE; APPLICATION-SOFTWARE object; MV
SYSTEM-SOFTWARE; SYSTEM-SOFTWARE object; MV

APPLICATION-SOFTWARE OBJECT

Application-Name; Application-names
Version; Version-number
NODE; NODE object

ACCESSORY OBJECT

Accessory-Name; Accessory-board-names
Accessory-Type; Accessory-board-types
I/O-Port; I/O-port-types
SETTING; SETTING object
NODE; NODE object

SETTING OBJECT

Switch-Setting; Switch-settings
Comment; Comments
ACCESSORY; ACCESSORY object

SYSTEM-SOFTWARE OBJECT

System-Software-Name; System-software-names
Version; Version-number
LAN; LAN object
NODE; NODE object; MV

CABLE-PLANT OBJECT

Item-Name; Item-name
Item-Quantity; Item-quantity
LAN; LAN object

SPARE-PARTS OBJECT

Spare-Name; Spare-part-name
Spare-Quantity; Spare-part-quantity
Location; Spare-part-location

DOMAIN DEFINITIONS

Accessory-board-names:

Text 10

Name of accessory boards/peripherals added to a node.
(i.e. Modem, Printer)

Accessory-board-types:

Text 10

Type of accessory boards added to a node.
(i.e. 2400 baud, PROPRINTER)

Application-names:

Text 15

Name of software application program.

Comments:

Text 200

Comments which explain the switch settings or
configuration of a particular add-in board.

Computer-node-names:

Text 6

(i.e. TN3)

CPU-model-numbers:

Text 10

Model number of the particular central processing unit.

CPU-serial-numbers:

Text 10

Serial number of the particular central processing unit.

Display-adapter-types:

Text 3

Type of display adapter card.
(i.e. EGA)

Display-model-numbers:

Text 10

Name and model number of monitor.

Expanded-memory:

Text 4

Amount of expanded memory installed.

Extended-memory:

Text 4

Amount of extended memory installed.

Floppy-disk-drive-number-one:

Text 12

Name of first disk drive.

(i.e. 1.2MB Floppy)

Floppy-disk-drive-number-two:

Text 12

Name of second disk drive (if any).

Hard-disk-number-one:

Text 4

Capacity of the first hard disk (if any).

(i.e. 20MB)

Hard-disk-number-two

Text 4

Capacity of the second hard disk (if any).

Item-name:

Text 20

Name of cable plant component.

(i.e. short distance kit)

I/O-port-types:

Text 10

Type of I/O port.

(i.e. Serial Port)

Item-quantity:

Numeric 2

Quantity of each cable plant component.

Keyboard-types:

Text 10

Type of keyboard.

LAN-identification:

Numeric 2

Specific 2 digit code

Location:

Text 10

Location of spare part.

(i.e. RM 158)

Motherboard-memory:

Text 4

Amount of RAM on the motherboard.
(i.e. 4MB)

Network-board:

Text 10

Type of network board.
(i.e. Token-Ring)

Spare-part-name:

Text 15

Name of spare part.

Spare-part-quantity:

Numeric 2

Quantity of a specific spare part.

Switch-settings:

Text 8, 0 or 1

Switch settings for add-in board. A 0 means off, 1 means on.

(i.e. 01101011)

System-software-names

Text 15

Name of network/microcomputer operating system

Version-number:

Text 4

Version number of application/system program

APPENDIX B

DATA FLOWS

Figure B.1 through B.4 are logical Data Flow Diagrams (DFD) which depict the essential functions of the LAN Configuration Management System. Each process, depicted by a circle, is uniquely identified by a number, i.e. 1.1, for IBM PC Network in Figure B.2. The DFD process numbers are the same as the numbers assigned the major process in the System Structure Charts shown in Figure A.1 through Figure A.3. Subprocesses are numbered such that the root process can be readily identified. For example, the subprocess, Add LAN Data is assigned the number 1.1.1. The '1.1' digits denote the root process, IBM PC Network, while the '.1' digit denotes that this is the first subprocess under IBM PC Network. The subsequent subprocess associated with Add LAN Data are denoted by an extension of the '1.1.1' number in the same manner (e.g. 1.1.1.1 for Add Node Data).

The following data flow diagrams depict the data flows for the IBM PC Network and Spare Parts functions of the system. The remaining system functions such as IBM Token Ring and TOPS Network use the same data flows for their respective networks. The diagrams were omitted to avoid redundancy.

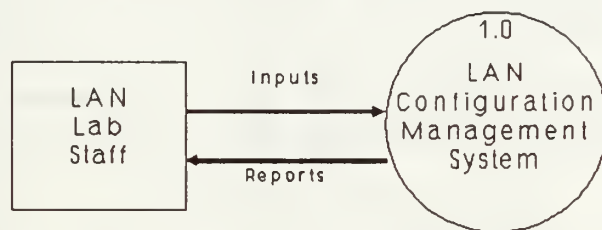


Figure B.1 Level 0 Data Flow Diagram

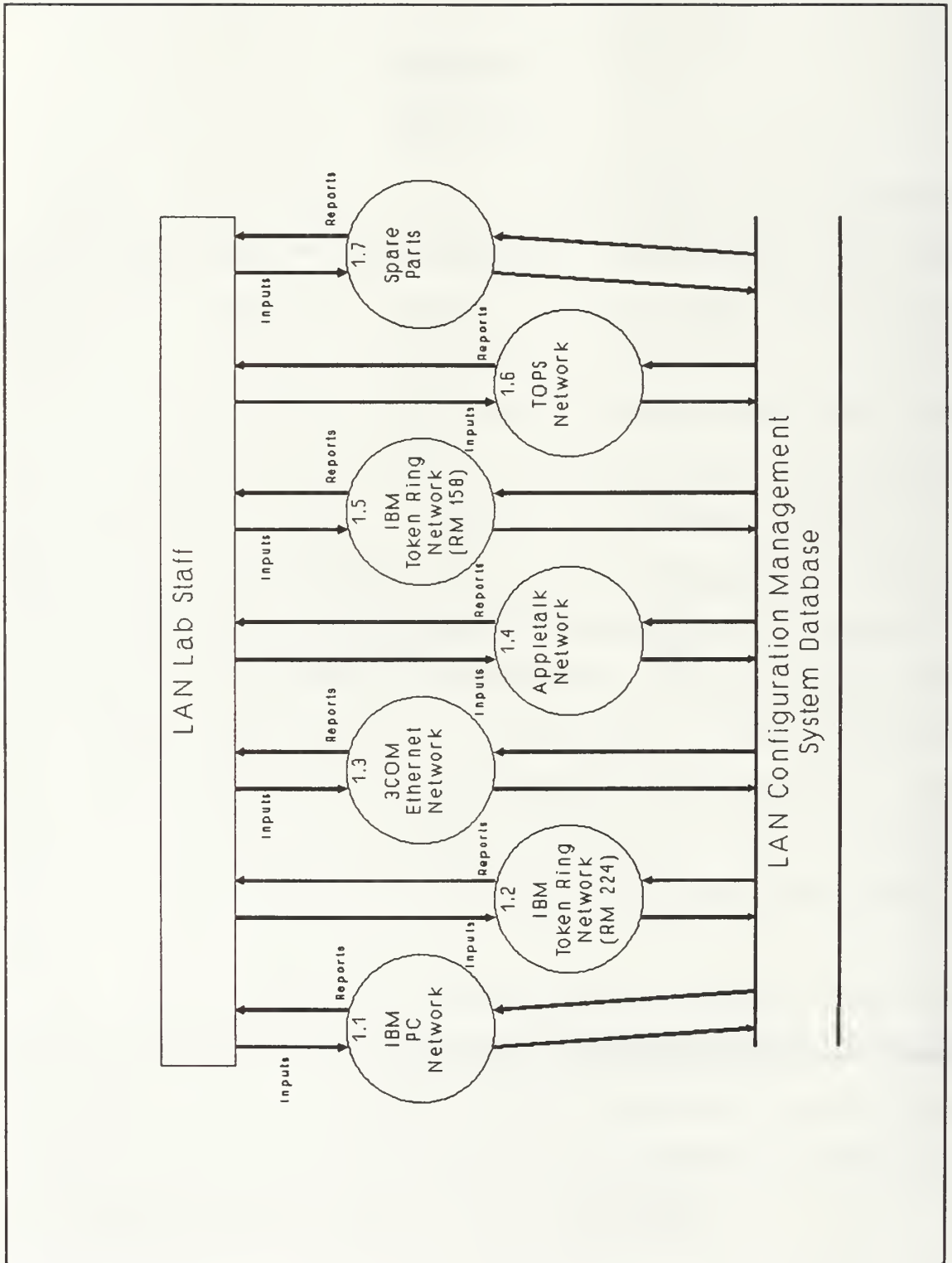


Figure B.2 Level 1 Data Flow Diagram

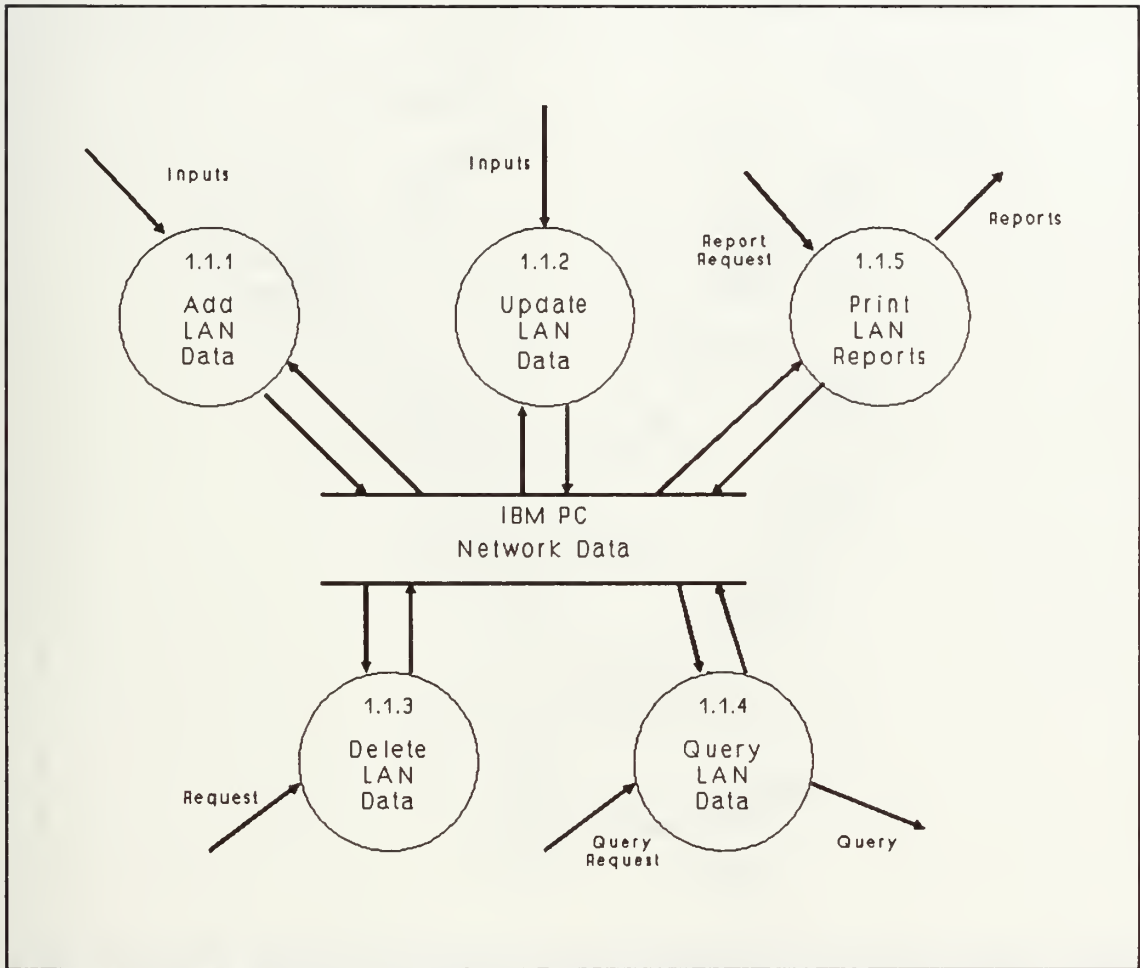


Figure B.3a IBM PC Network Data Flow Diagram

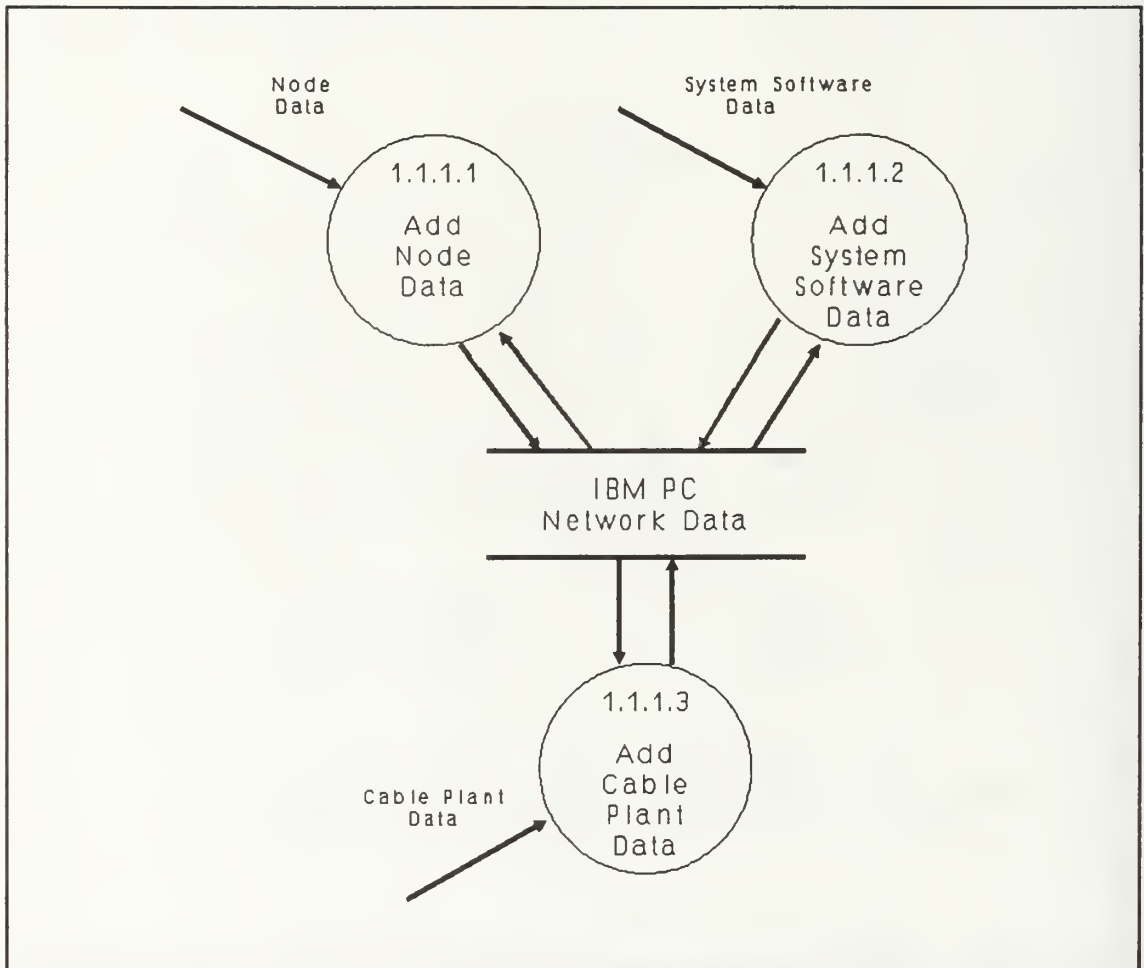


Figure B.3b IBM PC Network Data Flow Diagram (con't)

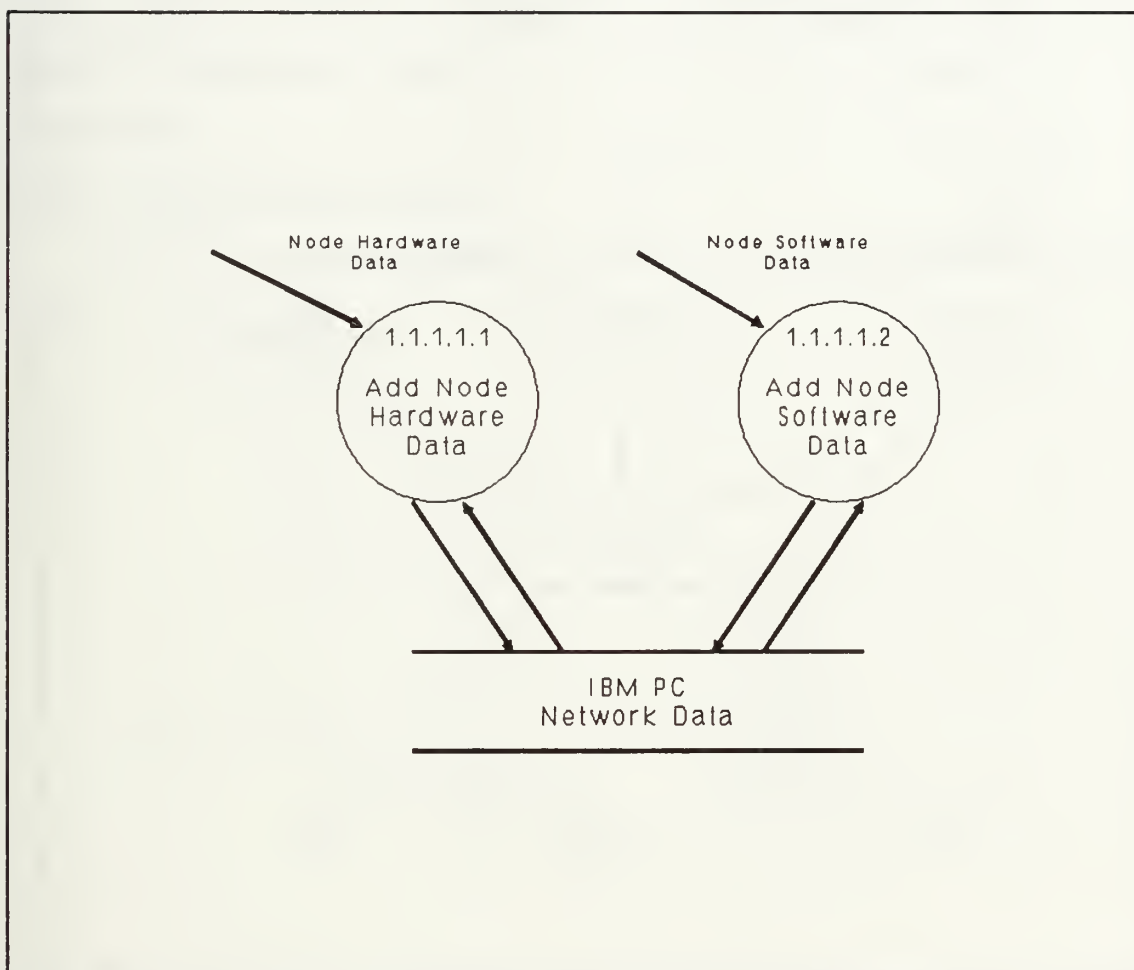


Figure B.3c IBM PC Network Data Flow Diagram (con't)

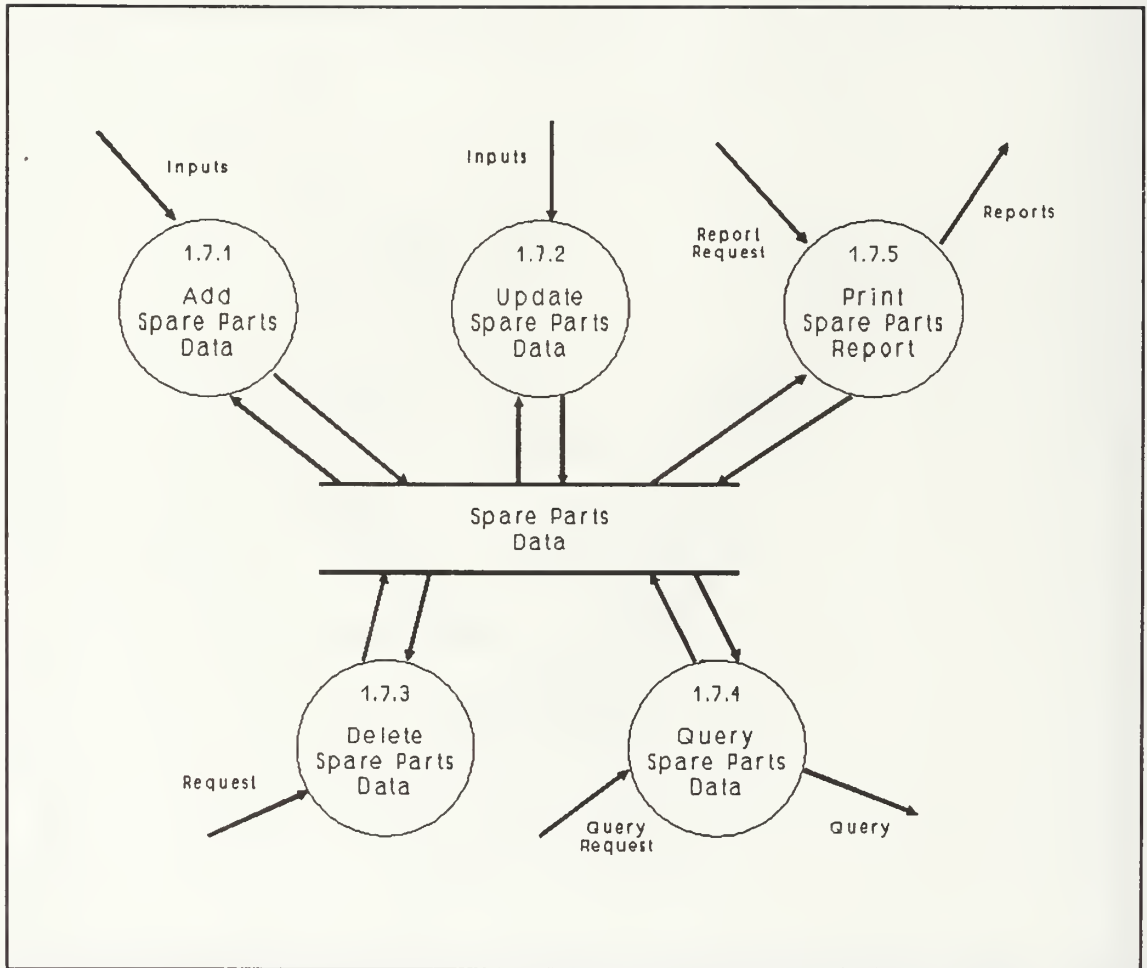


Figure B.4 Spare Parts Data Flow Diagram

APPENDIX C

FUNCTIONAL COMPONENT DESCRIPTIONS

This appendix contains the data form materializations, the add, update, delete and display mechanisms, and the special control mechanisms required to fulfill the requirements of the LAN staff.

DATA FORM MATERIALIZATIONS

Figure C.1 through Figure C.31 illustrate the data forms that are required for the system.

NETWORK NAME

NODE NAME: CPU MODEL #:

CPU SERIAL #: DISPLAY MODEL #:

DISPLAY ADAPTER TYPE: MOTHERBOARD MEMORY:

EXPANDED MEMORY: EXTENDED MEMORY:

HARD DISK 1: HARD DISK 2:

FLOPPY DISK DRIVE 1:

FLOPPY DISK DRIVE 2:

KEYBOARD TYPE: NETWORK BOARD:

DO YOU WANT TO ADD AN ACCESSORY? (Y/N): ☐

Figure C.1 Add Node Data Form

NETWORK NAME

NODE NAME:

ACCESSORY NAME:

ACCESSORY TYPE:

I/O PORT:

DO YOU WANT TO ADD A SWITCH SETTING AND/OR COMMENTS? (Y/N): ☐

DO YOU WANT TO ADD ANOTHER ACCESSORY? (Y/N): ☐

Figure C.2 Add Node Accessory Data Form

NETWORK NAME

NODE NAME:

ACCESSORY NAME:

SWITCH SETTING: ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

1 2 3 4 5 6 7 8

COMMENT:

Figure C.3 Add Node Accessory Setting Data Form

NETWORK NAME

NODE NAME:

APPLICATION NAME:

VERSION:

DO YOU WANT TO ADD ANOTHER SOFTWARE PACKAGE? (Y/N): ☐

Figure C.4 Add Node Software Data Form

NETWORK NAME

SYSTEM SOFTWARE NAME:

VERSION:

DO YOU WANT TO ADD MORE SYSTEM SOFTWARE? (Y/N): ☐

Figure C.5 Add System Software Data Form

NETWORK NAME

CABLE PLANT ITEM NAME:

ITEM QUANTITY:

DO YOU WANT TO ADD ANOTHER CABLE PLANT ITEM? (Y/N):

Figure C.6 Add Cable Plant Data Form

SPARE PARTS

SPARE NAME:

QUANTITY:

LOCATION:

DO YOU WANT TO ADD ANOTHER SPARE PART? (Y/N): ☐

Figure C.7 Add Spare Parts Data Form

NETWORK NAME

ENTER NODE NAME TO BE UPDATED:

NODE NAME: CPU MODEL #:

CPU SERIAL #: DISPLAY MODEL #:

DISPLAY ADAPTER TYPE: MOTHERBOARD MEMORY:

EXPANDED MEMORY: EXTENDED MEMORY:

HARD DISK 1: HARD DISK 2:

FLOPPY DISK DRIVE 1:

FLOPPY DISK DRIVE 2:

KEYBOARD TYPE: NETWORK BOARD:

DO YOU WANT TO UPDATE ACCESSORY DATA? (Y/N): ☐

Figure C.8 Update Node Data Form

NETWORK NAME

NODE NAME:

ACCESSORY NAME:

ACCESSORY TYPE:

I/O PORT:

DO YOU WANT TO UPDATE SWITCH SETTING AND/OR COMMENTS? (Y/N): ☐

DO YOU WANT TO UPDATE ANOTHER ACCESSORY? (Y/N): ☐

Figure C.9 Update Node Accessory Data Form

NETWORK NAME

NODE NAME:

ACCESSORY NAME:

SWITCH SETTING: ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

1 2 3 4 5 6 7 8

COMMENT:

Figure C.10 Update Node Accessory Setting Data Form

NETWORK NAME

ENTER NODE NAME OF SOFTWARE PACKAGE TO BE UPDATED:

NODE NAME:

APPLICATION NAME:

VERSION:

DO YOU WANT TO UPDATE ANOTHER SOFTWARE PACKAGE? (Y/N): ☐

Figure C.11 Update Node Software Data Form

NETWORK NAME

SYSTEM SOFTWARE NAME:

VEWRSION:

DO YOU WANT TO UPDATE MORE SYSTEM SOFTWARE? (Y/N): ☐

Figure C.12 Update System Software Data Form

NETWORK NAME

CABLE PLANT ITEM NAME:

ITEM QUANTITY:

DO YOU WANT TO UPDATE ANOTHER CABLE PLANT ITEM? (Y/N):

Figure C.13 Update Cable Plant Data Form

SPARE PARTS

SPARE NAME:

QUANTITY:

LOCATION:

DO YOU WANT TO UPDATE ANOTHER SPARE PART? (Y/N): ☐

Figure C.14 Update Spare Parts Data Form

NETWORK NAME

ENTER NAME OF NODE TO BE DELETED:

NODE NAME:

IS THIS THE NODE TO BE DELETED? (Y/N): ☐

DO YOU WANT TO DELETE AN ACCESSORY? (Y/N): ☐

Figure C.15 Delete Node Data Form

NETWORK NAME

ENTER NODE NAME OF ACCESSORY TO BE DELETED:

NODE NAME:

ACCESSORY NAME:

IS THIS THE ACCESSORY TO BE DELETED? (Y/N): ☐

Figure C.16 Delete Node Accessory Data Form

NETWORK NAME

ENTER NODE NAME OF SOFTWARE TO BE DELETED:

NODE NAME:

APPLICATION NAME:

IS THIS THE SOFTWARE APPLICATION TO BE DELETED? (Y/N): ☐

Figure C.17 Delete Node Software Data Form

NETWORK NAME

ENTER NAME OF SYSTEM SOFTWARE TO BE DELETED:

SYSTEM SOFTWARE NAME:

IS THIS THE SYSTEM SOFTWARE TO BE DELETED? (Y/N): ☐

Figure C.18 Delete System Software Data Form

NETWORK NAME

ENTER CABLE PLANT ITEM TO BE DELETED:

CABLE PLANT ITEM NAME:

IS THIS THE CABLE PLANT ITEM TO BE DELETED? (Y/N): ☐

Figure C.19 Delete Cable Plant Data Form

NETWORK NAME

ENTER SPARE PART NAME TO BE DELETED:

SPARE PART NAME:

IS THIS THE SPARE PART TO BE DELETED? (Y/N): ☐

Figure C.20 Delete Spare Parts Data Form

NETWORK NAME

ENTER NODE NAME TO BE QUERIED:

NODE NAME: CPU MODEL #:

CPU SERIAL #: DISPLAY MODEL #:

DISPLAY ADAPTER TYPE: MOTHERBOARD MEMORY:

EXPANDED MEMORY: EXTENDED MEMORY:

HARD DISK 1: HARD DISK 2:

FLOPPY DISK DRIVE 1:

FLOPPY DISK DRIVE 2:

KEYBOARD TYPE: NETWORK BOARD:

DO YOU WANT TO VIEW ASSOCIATED ACCESSORIES? (Y/N):

Figure C.21 Query Node Data

NETWORK NAME

NODE NAME:

ACCESSORY NAME:

ACCESSORY TYPE:

I/O PORT:

SWITCH SETTING: ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

1 2 3 4 5 6 7 8

COMMENT:

DO YOU WANT TO VIEW ANOTHER ACCESSORY? (Y/N): ☐

Figure C.22 Query Accessory and Switch Setting Data

NETWORK NAME

ENTER NODE NAME:

NODE NAME:

SYSTEM/ APPLICATION SOFTWARE	VERSION
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>

PRESS <ENTER> FOR ADDITIONAL SOFTWARE APPLICATIONS

Figure C.23 Query Node Software Data

NETWORK NAME

SYSTEM SOFTWARE

VERSION

████████████████████

██████

████████████████████

██████

PRESS <ENTER> TO CONTINUE

Figure C.24 Query System Software Data

NETWORK NAME	
<u>CABLE PLANT ITEM</u>	<u>QUANTITY</u>
██	██
██	██
██	██
██	██
██	██
██	██

PRESS <ENTER> TO CONTINUE

Figure C.25 Query Cable Plant Data

NETWORK NAME		
<u>SPARE PART NAME</u>	<u>QUANTITY</u>	<u>LOCATION</u>
████████████████████	██	████████████████
████████████████████	██	████████████████
████████████████████	██	████████████████
████████████████████	██	████████████████
████████████████████	██	████████████████
████████████████████	██	████████████████
████████████████████	██	████████████████

PRESS <ENTER> FOR ADDITIONAL SPARE PARTS

Figure C.26 Query Spare Parts Data

IBM TOKEN RING (RM 224)

NODE HARDWARE CONFIGURATION

Node Name: TN3
CPU Model #: 80286
CPU Serial #: 007246
Display Model #: 71D07589C
Display Adapter Type: EGA
Keyboard Type: Enhanced
Motherboard Memory: 640K
Expanded Memory: None
Extended Memory: None
Network Board: Token Ring
Hard Disk 1: 20MB
Hard Disk 2: N/A
Floppy Disk Drive 1: 1.2MB Floppy
Floppy Disk Drive 2: 360K Floppy
Accessories:
Name: Modem Type: 1200 Baud I/O Port: N/A
Name: Coprocessor Type: 80287 I/O Port: N/A
Name: Printer Type: PROPRINTERII I/O Port: Parallel

Figure C.27 Sample Node Hardware Configuration Output

IBM TOKEN RING (RM224)
NODE SOFTWARE CONFIGURATION

Node Name: TN3

<u>Software</u>	<u>Version</u>
PC LAN	1.12
MS DOS	3.3
LOTUS 123	2.01
WordPerfect	5.0
Harvard Graphics	2.1
DBASE III+	2.0
SmartCom	2.1
1DIR+	1.0

Figure C.28 Sample Node Software Configuration Output

IBM TOKEN RING (RM 224)
SYSTEM SOFTWARE CONFIGURATION

<u>SYSTEM SOFTWARE NAME</u>	<u>VERSION</u>
MS DOS	3.3
PC LAN	1.12

Figure C.29 Sample System Software Configuration Output

IBM TOKEN RING (RM 224)

CABLE PLANT CONFIGURATION

<u>Cable Plant Item</u>	<u>Quantity</u>
Shielded Twisted Pr	100ft
TR Adapter Cable	20
Patch Cable	10
MAU	4

Figure C.30 Sample Cable Plant Configuration Output

SPARE PARTS LISTING

<u>Item Name</u>	<u>Quantity</u>	<u>Location</u>
Hard Disk 20MB	6	RM 158
EGA Adapter	3	RM 158
Token Ring Brd	2	RM 158
PC Board	2	RM 224
3270 Emul Brd	2	RM 158
2400 Baud Modem	1	RM 158
Ethernet Board	2	RM 158
Hardcard 20MB	1	RM 158

Figure C.31 Sample Spare Parts Output

II. MECHANISMS

This section of Appendix C contains a summary of the add, update, and display mechanisms required for each of the data stores.

NODE Update Mechanisms

- D. Add NODE data
 - 1. Inputs
 - Individual node data
 - 2. Outputs
 - New NODE object instance in database
 - 3. Processing notes
 - None
- E. Update NODE data
 - 1. Inputs
 - NODE object from database
 - Changed individual node data
 - 2. Outputs
 - Modified NODE object instance in database
 - 3. Processing notes
 - None
- F. Delete NODE data
 - 1. Inputs
 - LAN staff request to remove node from network
 - 2. Outputs
 - Confirmation notice on screen
 - 3. Processing notes
 - None

ACCESSORY Update Mechanisms

- I. Add ACCESSORY data
 - A. Inputs
 - NODE data from database
 - B. Outputs
 - New ACCESSORY object instance in database
 - C. Processing notes
 - User will be prompted to add switch settings
- II. Update ACCESSORY data
 - A. Inputs
 - ACCESSORY data from database
 - Changed individual accessory data

- B. Outputs
 - Modified ACCESSORY object instance in database
- C. Processing notes
 - User will be prompted for switch settings update

III. Delete ACCESSORY data

- A. Inputs
 - LAN staff request to remove accessory
- B. Outputs
 - Confirmation Notice on screen
- C. Processing notes
 - Done at same time as deletion of a node

SETTING Update Mechanisms

I. Add SETTING data

- A. Inputs
 - ACCESSORY data from database
 - Individual switch setting data
 - Comments
- B. Outputs
 - New SETTING object instance in database
- C. Processing notes
 - None

II. Update SETTING data

- A. Inputs
 - SETTING data from database
 - Changed individual setting data
- B. Outputs
 - Modified SETTING object instance in database
- C. Processing notes
 - None

III. Delete SETTING data

- A. Inputs
 - None
- B. Outputs
 - None
- C. Processing notes
 - Done at same time as deletion of node/accessory

APPLICATION-SOFTWARE Update Mechanisms

I. Add APPLICATION-SOFTWARE data

- A. Inputs
 - Software application data

- B. Outputs
 - New APPLICATION-SOFTWARE object instance in database
- C. Processing notes
 - None

II. Update APPLICATION-SOFTWARE data

- A. Inputs
 - APPLICATION-SOFTWARE object from database
 - Changed individual software data
- B. Outputs
 - Modified APPLICATION-SOFTWARE object instance in database
- C. Processing notes
 - None

III. Delete APPLICATION-SOFTWARE data

- A. Inputs
 - Name of software application to be deleted
- B. Outputs
 - Confirmation notice on screen
- C. Processing notes
 - Done at the same time as deletion of a node

SYSTEM-SOFTWARE Update Mechanisms

I. Add SYSTEM-SOFTWARE data

- A. Inputs
 - Individual system software data
- B. Outputs
 - New SYSTEM-SOFTWARE object instance in database
- C. Processing notes
 - None

II. Update SYSTEM SOFTWARE data

- A. Inputs
 - SYSTEM-SOFTWARE object from database
 - Changed individual system software data
- B. Outputs
 - Modified SYSTEM-SOFTWARE object instance in database
- C. Processing notes
 - None

III. Delete SYSTEM-SOFTWARE data

- A. Inputs
 - LAN staff request to remove system software from Network

- B. Outputs
 - Confirmation notice on screen
- C. Processing notes
 - None

CABLE-PLANT Update Mechanisms

- I. Add CABLE-PLANT data
 - A. Inputs
 - Individual cable plant item data
 - B. Outputs
 - New CABLE-PLANT object instance in database
 - C. Processing notes
 - None
- II. Update CABLE-PLANT data
 - A. Inputs
 - CABLE-PLANT object from database
 - Changed individual cable plant data
 - B. Outputs
 - Modified CABLE-PLANT object instance in database
 - C. Processing notes
 - None
- III. Delete CABLE-PLANT data
 - A. Inputs
 - LAN staff request to remove cable plant item from Network
 - B. Outputs
 - Confirmation notice on screen
 - C. Processing notes
 - None

SPARE-PARTS Update Mechanisms

- I. Add SPARE-PARTS data
 - A. Inputs
 - Individual spare parts data
 - B. Outputs
 - New SPARE-PARTS object instance in database
 - C. Processing notes
 - None
- II. Update SPARE-PARTS data
 - A. Inputs
 - SPARE-PARTS object from database
 - Changed individual spare parts data
 - B. Outputs
 - Modified SPARE-PARTS object instance in database

C. Processing notes

- None

III. Delete SPARE-PARTS data

A. Inputs

- LAN staff request to remove spare part from database

B. Outputs

- Confirmation notice on screen

C. Processing notes

- None

NODE Display Mechanisms

I. Query NODE data

A. Output description

- Screen showing node data for a specified node

B. Source data

- NODE object

C. Processing notes

- Processed by the LAN staff on demand

II. Query ACCESSORY and SETTING data

A. Output description

- Screen showing accessory and switch setting for a specified node

B. Source data

- ACCESSORY object
- SETTING object

C. Processing notes

- Processed by the LAN staff on demand

III. Query Node Software data

A. Output description

- Screen showing system and application software loaded on a specified node.

B. Source data

- APPLICATION-SOFTWARE object
- SYSTEM-SOFTWARE object

C. Processing notes

- Processed by the LAN staff on demand

IV. Node Hardware Configuration Report

A. Output description

- Report showing all NODE and ACCESSORY data for a particular node

B. Source data

- NODE object
- ACCESSORY object

- C. Processing notes
 - Processed by the LAN staff on demand

V. Node Software Configuration Report

- A. Output description
 - Report showing all system and application software loaded on a particular node
- B. Source data
 - APPLICATION-SOFTWARE object
 - SYSTEM-SOFTWARE object
- C. Processing notes
 - Processed by the LAN staff on demand

SYSTEM SOFTWARE Display Mechanisms

I. Query SYSTEM-SOFTWARE data

- A. Output description
 - Screen showing system software loaded on a network
- B. Source data
 - SYSTEM-SOFTWARE object
- C. Processing notes
 - Processed by LAN staff on demand

II. System Software Configuration Report

- A. Output description
 - Report showing all system software loaded on a particular network
- B. Source data
 - SYSTEM-SOFTWARE object
- C. Processing notes
 - Processed by the LAN staff on demand

CABLE-PLANT Display Mechanisms

I. Query CABLE-PLANT data

- A. Output description
 - Screen showing cable plant items attached to a network
- B. Source data
 - CABLE-PLANT object
- C. Processing notes
 - Processed by LAN staff on demand

II. Cable Plant Configuration Report

- A. Output description
 - Report showing all cable plant items attached to a particular network

- B. Source data
 - CABLE-PLANT object
- C. Processing notes
 - Processed by the LAN staff on demand

SPARE-PARTS Display Mechanisms

- I. Query SPARE-PARTS data
 - A. Output description
 - Screen showing spare parts items in inventory
 - B. Source data
 - SPARE-PARTS object
 - C. Processing notes
 - Processed by LAN staff on demand
- II. Spare Parts Configuration Report
 - A. Output description
 - Report showing all spare parts items in inventory
 - B. Source data
 - SPARE-PARTS object
 - C. Processing notes
 - Processed by the LAN staff on demand

APPENDIX D

APPLICATION DESIGN COMPONENTS

Within the following pages the reader is provided with various items to assist in understanding the application design of the LAN Configuration Management System. The contents of this appendix are:

1. System Relation Diagram
2. System Relation Definitions
3. System Menu Hierarchy
4. System Application Logic

Figure D.1 depicts the relation of the different objects involved in the system database to each another. The circle on the lines that connect the objects indicates an optional relationship that exists between the two. A short hash mark on the line indicates that a mandatory relationship exists. A short leg attached to the end of a line indicates that multiple instances of that object may exist. The mark closest to the object box would indicate the relationship that exists between that box and the box that the line connects. For example, the relationship between the LAN object and the NODE is optional (indicated by the circle). The LAN does not require the NODE. On the other hand, the relationship between the NODE object and the LAN object is mandatory (indicated by

the hash mark). The NODE is required to be attached to a LAN.
The underlined words within the boxes indicate that they are
a key for that particular relation.

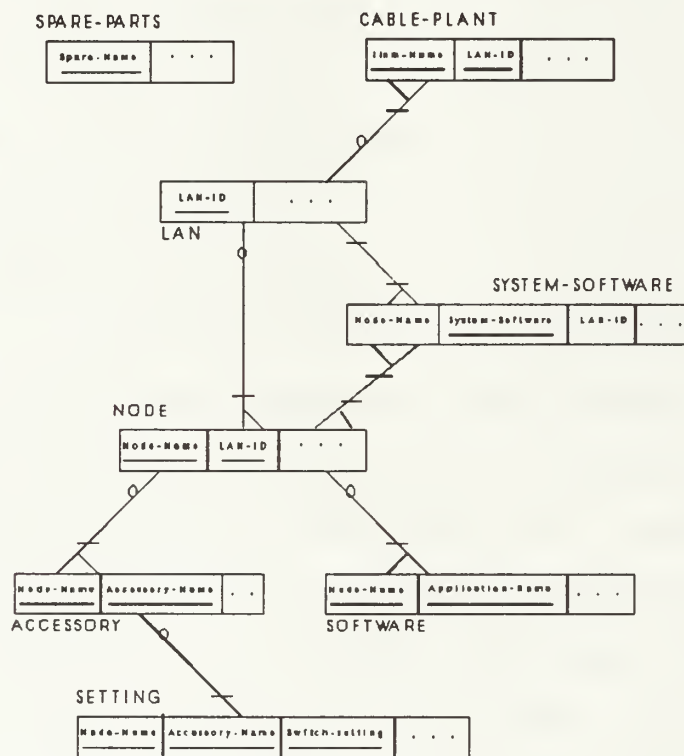


Figure D.1 System Relation Diagram

RELATION DEFINITIONS

LAN		
Item	Length	Type
LAN-ID	2	N

NODE		
Item	Length	Type
Node-Name	6	A
CPU-Model-#	10	A/N
CPU-Serial-#	10	A/N
Display-Model-#	10	A/N
Display-Adapter-Type	3	A
Keyboard-Type	10	A
Motherboard-Memory	4	A/N
Expanded-Memory	4	A/N
Extended-Memory	4	A/N
Network-Board	10	A
Hard-Disk-1	4	A/N
Hard-Disk-2	4	A/N
Floppy-Disk-Drive-1	12	A/N
Floppy-Disk-Drive-2	12	A/N

ACCESSORY		
Item	Length	Type
Accessory-Name	10	A
Accessory-Type	10	A/N
I/O-Port	10	A/N

SETTING		
Item	Length	Type
Switch-Setting Comment	8 200	N A/N

APPLICATION-SOFTWARE		
Item	Length	Type
Application-Name Version	15 4	A/N A/N

SYSTEM-SOFTWARE		
Item	Length	Type
System-Software-Name Version	15 4	A A/N

CABLE-PLANT		
Item	Length	Type
Item-Name Item-Quantity	20 2	A/N N

SPARE-PARTS		
Item	Length	Type
Spare-Name Spare-Quantity Location	15 2 10	A/N N A/N

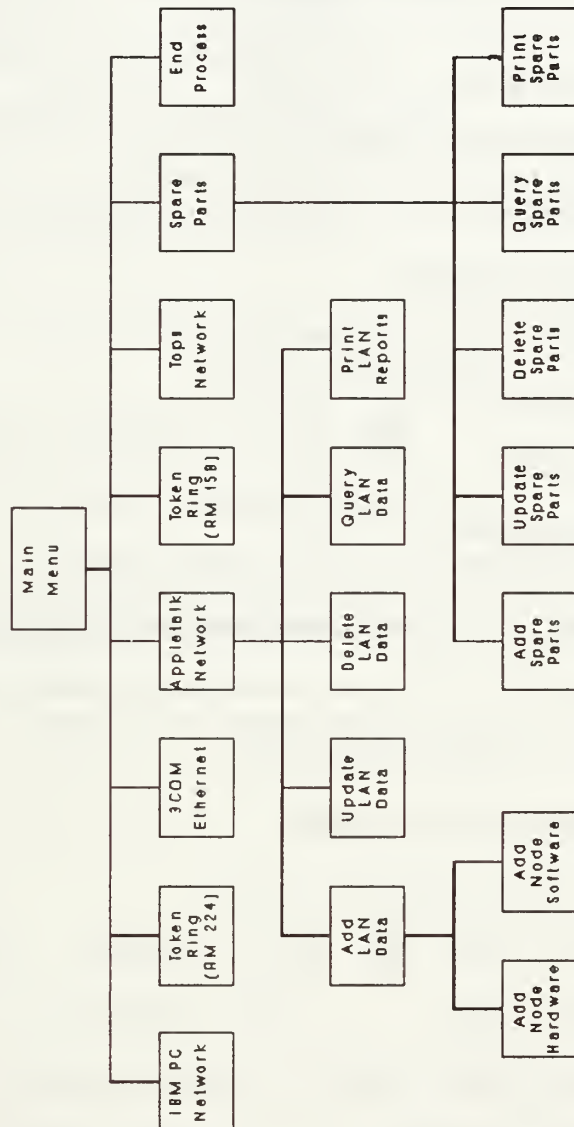


Figure D.2 System Hierarchy of Menus

LAN CONFIGURATION MANAGEMENT
SYSTEM

MAIN MENU

SELECT DESIRED NETWORK: ☐

A = IBM PC NETWORK

B = IBM TOKEN RING NETWORK (RM 224)

C = 3COM ETHERNET

D = APPLETALK NETWORK

E = IBM TOKEN RING NETWORK (RM 158)

F = TOPS NETWORK

G = SPARE PARTS

H = END PROCESSING

Figure D.3 LAN CONFIGURATION MANAGEMENT SYSTEM Main Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = ADD LAN DATA

B = UPDATE LAN DATA

C = DELETE LAN DATA

D = QUERY LAN DATA

E = PRINT LAN REPORTS

F = RETURN TO MAIN MENU

Figure D.4 Sample Network Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = ADD NODE DATA

B = ADD SYSTEM SOFTWARE DATA

C = ADD CABLE PLANT DATA

D = RETURN TO PREVIOUS MENU

Figure D.5 Sample Add Network Data Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = ADD NODE HARDWARE DATA

B = ADD NODE APPLICATION SOFTWARE DATA

C = RETURN TO PREVIOUS MENU

Figure D.6 Sample Add Node Hardware/Software Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = UPDATE NODE DATA

B = UPDATE SYSTEM SOFTWARE DATA

C = UPDATE CABLE PLANT DATA

D = RETURN TO PREVIOUS MENU

Figure D.7 Sample Update Network Data Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = UPDATE NODE HARDWARE DATA

B = UPDATE MODE APPLICATION SOFTWARE DATA

C = RETURN TO PREVIOUS MENU

Figure D.8 Sample Update Node Hardware/Software Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = DELETE NODE DATA

B = DELETE SYSTEM SOFTWARE DATA

C = DELETE CABLE PLANT DATA

D = RETURN TO PREVIOUS MENU

Figure D.9 Sample Delete Network Data Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = DELETE NODE HARDWARE DATA

B = DELETE NODE APPLICATION SOFTWARE DATA

C = RETURN TO PREVIOUS MENU

Figure D.10 Sample Delete Node Hardware/Software Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = QUERY NODE DATA

B = QUERY SYSTEM SOFTWARE DATA

C = QUERY CABLE PLANT DATA

D = RETURN TO PREVIOUS MENU

Figure D.11 Sample Query Network Data Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = QUERY NODE HARDWARE DATA

B = QUERY NODE SOFTWARE DATA

C = RETURN TO PREVIOUS MENU

Figure D.12 Sample Query Node Hardware/Software Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = PRINT NODE CONFIGURATION

B = PRINT SYSTEM SOFTWARE CONFIGURATION

C = PRINT CABLE PLANT CONFIGURATION

D = RETURN TO PREVIOUS MENU

Figure D.13 Sample Print Network Configuration Menu

IBM PC NETWORK

SELECT DESIRED OPTION: ☐

A = PRINT NODE HARDWARE CONFIGURATION

B = PRINT NODE SOFTWARE CONFIGURATION

C = RETURN TO PREVIOUS MENU

Figure D.14 Sample Print Node Hardware/Software Menu

SPARE PARTS MENU

SELECT DESIRED OPTION: ☐

A = ADD SPARE PARTS

B = UPDATE SPARE PARTS

C = DELETE SPARE PARTS

D = QUERY SPARE PARTS

E = PRINT SPARE PARTS REPORT

F = RETURN TO MAIN MENU

Figure D.15 Sample Spare Parts Menu

APPLICATION LOGIC

I. Low Level Menu Choice:

Add Node Hardware Data

Materialization:

See Figure C.1

See Figure C.2

See Figure C.3

Logic:

Display input form.

Position cursor at first field.

Enter key moves cursor through fields (allow blank input for any field except Node Name).

Upon completion of data entry, write record to file, then prompt to add Accessory data. If yes, then create a new Accessory instance with Node Name.

If no, then exit to Add Network Data Menu.

Display Accessory input form.

Position cursor at second field.

Enter key moves through fields.

Upon completion of data entry, write record to file, then prompt for Switch Setting data. If yes, then create a new Setting instance with Node Name and Accessory name.

Display Setting input form.

Position cursor at third field.

Enter key moves through fields.

Upon completion of data entry, write record to file, then prompt to add another Accessory.

If no, then prompt to add another Accessory.

If yes, then create a new Accessory with node name.

If no, then exit to Add Network Data Menu.

Integrity/Security Considerations:

A Node Name must be entered for each new record to insure the NODE, ACCESSORY and SETTING relations maintain their integrity. Entry of a blank Node name will cause this option to terminate and return to the main menu.

II. Low Level Menu Choice:

Add Node Application Software Data

Materialization:

See Figure C.4

Logic:

Display input form.

Position cursor at first field.

Enter key moves through fields.

Upon completion of data entry, write record to file,
then prompt for entry of another software package.

Integrity/Security Considerations:

A Node Name must be entered for each application.
Entry of a blank Node Name will cause this option
to terminate and return to main menu.

III. Low Level Menu Choice:

Add System Software Data

Materialization:

See Figure C.5

Logic:

Display input form.

Position cursor at first field.

Enter key moves through fields.

Upon completion of data entry, write record to file,
then prompt for entry of another peripheral.

Integrity/Security Considerations:

None.

IV. Low Level Menu Choice:

Add Cable Plant Data

Materialization:

See Figure C.6

Logic:

Display input form.

Position cursor at first field.

Enter key moves through fields.

Upon completion of data entry, write record to file,
then prompt for entry of another Cable Plant item.

Integrity/Security Considerations:

None.

V. Low Level Menu Choice:

Add Spare Parts Data

Materialization:

See Figure C.7

Logic:

Display input form.

Position cursor at first field.

Enter key moves through fields.

Upon completion of data entry, write record to file,
then prompt for entry of another Spare Parts item.

Integrity/Security Considerations:

None.

VI. Low Level Menu Choice:

Update Node Hardware Data

Materialization:

See Figure C.8

See Figure C.9

See Figure C.10

Logic:

Prompt for Node Name of record to be updated.
Display update form with current information.
Position cursor at first field.
Enter Key moves cursor through fields.
Only data that is rekeyed is changed
Upon completion of update write record to file,
prompt for update to Accessory Data. If yes,
then display Accessory update form with current
information.
Position cursor at first field.
Enter key moves cursor through fields.
Only data that is rekeyed is changed.
Upon completion of update write record to file,
then prompt for update to Setting Data.
If no, then prompt to update Setting data.
If yes, then display Setting update form with current
information.
Position cursor at first field.
Enter key moves cursor through fields.
Only data that is rekeyed is changed.
If no, then exit to Update Network Data Menu.

Integrity/Security Considerations:

Do not allow changes to be made to Node Name field.
A change to Node Name will allow the NODE, ACCESSORY
and SETTING relations to become un-synchronized. Any
change to Node Name must be accomplished by deleting
the record and recreating it.

VII. Low Level Menu Choice:

Update Node Application Software Data

Materialization:

See Figure C.11

Logic:

Prompt for Node Name of record to be updated.
Display Update form with current information.
Position cursor at first field.
Only data that is rekeyed is changed.
Upon completion of update write record to file,
then prompt for update to another software
package.

Integrity/Security Considerations:

None.

VIII. Low Level Menu Choice:

Update System Software Data

Materialization:

See Figure C.12

Logic:

Display Update form with current information.
Position cursor at first field.
Only data that is rekeyed is changed.
Upon completion of update write record to file,
then prompt for update to other system
software.

Integrity/Security Considerations:

None.

IX. Low Level Menu Choice:

Update Cable Plant Data

Materialization:

See Figure C.13

Logic:

Display Update form with current information.
Position cursor at first field.
Only data that is rekeyed is changed.
Upon completion of update write record to file,
then prompt for update to another cable plant item.

Integrity/Security Considerations:

None.

X. Low Level Menu Choice:

Update Spare Parts Data

Materialization:

See Figure C.14

Logic:

Display Update form with current information.
Position cursor at first field.
Only data that is rekeyed is changed.
Upon completion of update write record to file,
then prompt for update to another spare parts item.

Integrity/Security Considerations:

None.

XI. Low Level Menu Choice:

Delete Node Hardware Data

Materialization:

See Figure C.15

See Figure C.16

Logic:

Prompt for Node Name to be deleted.
Display delete form with current information.
Verify for proper record to delete.
If correct record, delete instance in NODE,
ACCESSORY and SETTING relations.
Prompt for Accessory deletion.
If yes, then display delete form with current
information.
Verify for proper record to delete.
If correct record, delete instance in ACCESSORY
and SETTING relations.
If no, then exit to Delete Network Data Menu.

Integrity/Security Considerations:

Insure that deletions are accomplished for the NODE,
ACCESSORY and SETTING files accordingly.

XII. Low Level Menu Choice:

Delete Node Application Software Data

Materialization:

See Figure C.17

Logic:

Prompt for Node Name of software package to be
deleted.
Display delete form with current information.
Verify for proper record to delete.
If wrong record then get next record.
If correct record, delete instance in
APPLICATION-SOFTWARE relation.

Integrity/Security Considerations:

Insure that deletion is accomplished for the APPLICATION-SOFTWARE file.

XIII. Low Level Menu Choice:

Delete System Software Data

Materialization:

See Figure C.18

Logic:

Display delete form with current information.
Verify for proper record to delete.
If wrong record then get next record.
If correct record, delete instance in SYSTEM-SOFTWARE relation.

Integrity/Security Considerations:

Insure that deletion is accomplished for the SYSTEM-SOFTWARE file.

XIV. Low Level Menu Choice:

Delete Cable Plant Data

Materialization:

See Figure C.19

Logic:

Display delete form with current information.
Verify for proper record to delete.
If wrong record then get next record.
If correct record, delete instance in CABLE-PLANT relation.

Integrity/Security Considerations:

Insure that deletion is accomplished for the CABLE-PLANT file.

XV. Low Level Menu Choice:

Delete Spare Parts

Materialization:

See Figure C.20

Logic:

Display delete form with current information.
Verify for proper record to delete.
If wrong record then get next record.
If correct record, delete instance in SPARE-PARTS
relation.

Integrity/Security Considerations:

Insure that deletion is accomplished for the
SPARE-PARTS file.

XVI. Low Level Menu Choice:

Query Node Hardware Data

Materialization:

See Figure C.21
See Figure C.22

Logic:

Prompt for Node Name to be queried.
Display Node Query form with current information.
Prompt to query associated accessories.
If yes, then display Accessory Query form.
Prompt to see additional accessories.
If no, then exit to Query Network Data menu.

Integrity/Security Considerations:

None

XVII. Low Level Menu Choice:

Query Node Software Data

Materialization:

See Figure C.23

Logic:

Prompt for Node Name of Software to be queried.
Display Software Query form with current information.
Return will display additional software that cannot
fit on one screen.

Integrity/Security Considerations:

None

XVIII. Low Level Menu Choice:

Query System Software Data

Materialization:

See Figure C.24

Logic:

Display System Software Query form with current
information.

Integrity/Security Considerations:

None

XIX. Low Level Menu Choice:

Query Cable Plant Data

Materialization:

See Figure C.25

Logic:

Display Cable Plant Query form with current information.

Integrity/Security Considerations:

None

XX. Low Level Menu Choice:

Query Spare Parts

Materialization:

See Figure C.26

Logic:

Display Spare Parts Query form with current information.

Integrity/Security Considerations:

None

XXI. Low Level Menu Choice:

Print Node Hardware Configuration

Materialization:

Figure C.27

Logic:

Get Node information from NODE relation.

Get Accessory information from ACCESSORY relation
based on the Node Name in the in the NODE
relation.

Print report in form shown in the materialization.

Integrity/Security Considerations:

None.

XXII. Low Level Menu Choice:

Print Node Software Configuration

Materialization:

Figure C.28

Logic:

Get Node information from NODE relation.

Get Software information from APPLICATION-SOFTWARE
and SYSTEM-SOFTWARE relation based on the Node
Name in the NODE relation.

Print report in form shown in the materialization.

Integrity/Security Considerations:

None.

XXIII. Low Level Menu Choice:

Print System Software Configuration

Materialization:

Figure C.29

Logic:

Get System Software information from SYSTEM-SOFTWARE relation.

Print report in form shown in the materialization.

Integrity/Security Considerations:

None.

XXIV. Low Level Menu Choice:

Print Cable Plant Configuration

Materialization:

Figure C.30

Logic:

Get Cable Plant information from CABLE-PLANT relation.
Print report in form shown in the materialization.

Integrity/Security Considerations:

None.

XXV. Low Level Menu Choice:

Print Spare Parts Configuration

Materialization:

Figure C.31

Logic:

Get Spare Parts information from SPARE-PARTS relation.
Print report in form shown in the materialization.

Integrity/Security Considerations:

None.

LIST OF REFERENCES

1. Superintendent, Naval Postgraduate School, NAVPGSCOLINST 5239.1, "ADP Security Plan," pp. 8.1-8.3 June 30, 1989.
2. Schneidewind, Norman F., "Software Maintenance: the Need for Standardization," Proceceedings of the IEEE, April, 1989, p. 9.
3. Kroenke, David M. and Dolan, Kathleen A., Database Processing: Fundamentals, Design, Implementation, Science Research Associates, Inc., 1988.
4. Hurwicz, Michael, "Network Management Tools," Computer World, pp. 85-90, May 1, 1989.
5. Townsend, Carl, Mastering dBASE III Plus, SYBEX, Alameda, California, 1986.

BIBLIOGRAPHY

"Guide to Software Configuration Management," Software Engineering Standards Subcommittee on Software Engineering of the IEEE Computer Society, March, 1987.

Jones, D. and Lee, D., "Managing Local Area Networks," Journal of Systems Management, July 1, 1987, pp. 32-34.

Madron, Thomas W., Local Area Networks: The Second Generation, John Wiley & Sons, Inc., 1988.

Samaras, Thomas T. and Czerwinski, Frank L., Fundamentals of Configuration Management, John Wiley & Sons, Inc., 1971.

Schneidewind, Norman F., "The of Software Maintenance," IEEE Transactions on Software Engineering, vol. SE-13, no. 3, March, 1987.

Stallings, William, Local Networks: An Introduction, Macmillin Publishing Company, 1984, pp. 319-324.

Whitten, Jeffery L., Bently, Lonnie D., and Ho, Thomas I.M., Systems Analysis & Design Methods, Times Mirror/Mosby College, 1986.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
Cameron Station
Alexandria, Virginia 22304-6145
2. Superintendent 2
Attn: Library, Code 0142
Naval Postgraduate School
Monterey, California 93943-5000
3. Commandant of the Marine Corps 2
Code TE06
Headquarters, U.S. Marine Corps
Washington, D.C. 20360-0001
4. Norman F. Schneidewind 2
Code 54Ss
Department of Administrative Sciences
Naval Postgraduate School
Monterey, California 93943-5000
5. Leon R. Sahlman 1
Code 54Sl
Department of Administrative Sciences
Naval Postgraduate School
Monterey, California 93943-5000
6. Captain Douglas A. Suriano 1
10 Larkwood Court
Stafford, Virginia 22554

1-19-588

Thesis

SS6402 Suriano

c.1 The design of a Local
Area Network Configura-
tion Management System
for the Naval Postgradu-
ate School Administrative
Sciences Department.



The design of a Local Area Network Confi



3 2768 000 86737 8

DUDLEY KNOX LIBRARY